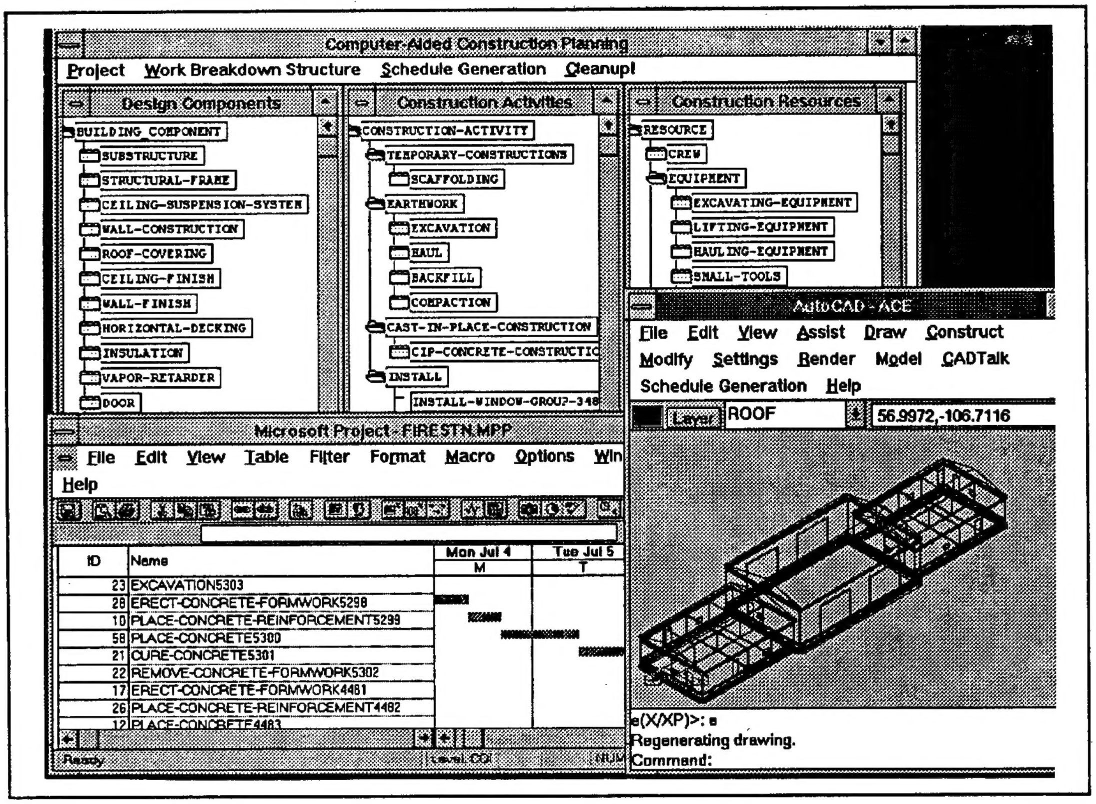
US Army Corps of Engineers
Construction Engineering Research Laboratories

USACERL Technical Report 96/82
August 1996

Integrating Object-Oriented CAD and Rule-Based Technologies for Construction Planning

by
Rajaram Ganeshan, Annette Stumpf, Sangyoon Chin, Liang Liu, and Bill Harrison



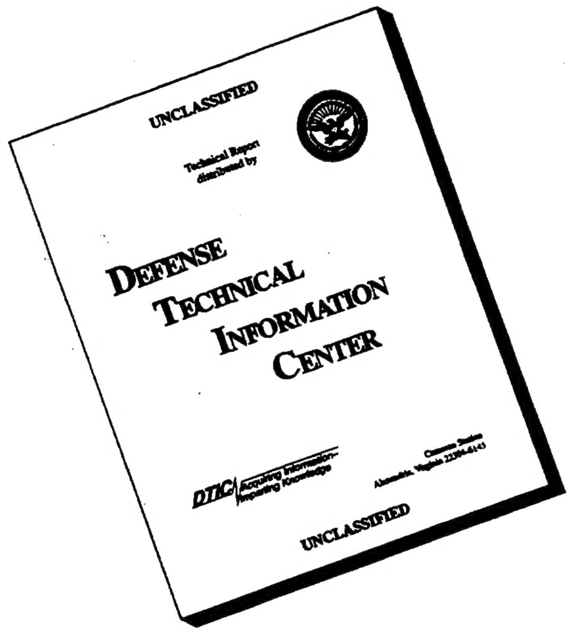
This report presents a methodology incorporating object-oriented and rule-based concepts to generate a preliminary construction plan for facility designs to: (1) compare alternative designs from a construction time and cost perspective; (2) animate the schedule to verify the schedule and identify constructibility problems before construction begins; (3) provide a good baseline schedule and cost estimate for the evaluation of contractor bids; and (4) determine the impact on schedule and costs due to modifications during the construction management phase. A prototype implemented in a LISP-based environment includes: (1) object-oriented models of entities relevant to construction planning; (2) knowledge-bases to generate activities, identify construction methods, and sequence activities; (3) interactive capability to control the level of detail in schedule by grouping components based on spatial location by construction zones; (4) interface with MCACES (Micro-Computer Aided Cost Engineering Support system) databases to obtain crew definitions, unit costs and productivity information for common construction activities; (5) interface with Microsoft® Project to do Critical Path Method analysis and display and manipulate schedule information; (6) capability to animate any portion of the schedule; and (7) capability to deal with incomplete design information by using construction solutions from previous projects. The prototype system was well received by schedulers and estimators.

19960903 090

DTIC QUALITY INSPECTED 1

Approved for public release; distribution is unlimited.

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

The contents of this report are not to be used for advertising, publication or promotional purposes. Citation of trade names does not constitute ar official endorsement or approval of the use of such commercial products. The findings of this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

DESTROY THIS REPORT WHEN IT IS NO LONGER NEEDED

DO NOT RETURN IT TO THE ORIGINATOR

USER EVALUATION OF REPORT

REFERENCE: USACERL Technical Report 96/82, *Integrating Object-Oriented CAD and Rule-Based Technologies for Construction Planning*

Please take a few minutes to answer the questions below, tear out this sheet, and return it to USACERL. As user of this report, your customer comments will provide USACERL with information essential for improving future reports.

1. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which report will be used.)

____

____

____

2. How, specifically, is the report being used? (Information source, design data or procedure, management procedure, source of ideas, etc.)

____

____

3. Has the information in this report led to any quantitative savings as far as manhours/contract dollars saved, operating costs avoided, efficiencies achieved, etc.? If so, please elaborate.

____

____

4. What is your evaluation of this report in the following areas?

a. Presentation: ______________________________

b. Completeness: ______________________________

c. Easy to Understand: ______________________________

d. Easy to Implement: ______________________________

e. Adequate Reference Material: ______________________________

f. Relates to Area of Interest: ______________________________

g. Did the report meet your expectations? ______________________________

h. Does the report raise unanswered questions? ______________________________

i. General Comments. (Indicate what you think should be changed to make this report and future reports of this type more responsive to your needs, more usable, improve readability, etc.)

____

____

____

____

____

____

5. If you would like to be contacted by the personnel who prepared this report to raise specific questions or discuss the topic, please fill in the following information.

Name: ______________________________

Telephone Number: ______________________________

Organization Address: ______________________________

____

____

6. Please mail the completed form to:

Department of the Army
CONSTRUCTION ENGINEERING RESEARCH LABORATORIES
ATTN: CECER-TR-I
P.O. Box 9005
Champaign, IL 61826-9005

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave Blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	August 1996	Final

4. TITLE AND SUBTITLE

Integrating Object-Oriented CAD and Rule-Based Technologies for Construction Planning

5. FUNDING NUMBERS

4A162784
AT41
FF-AS4

6. AUTHOR(S)

Rajaram Ganeshan, Annette Stumpf, Sangyoon Chin, Liang Liu, and Bill Harrison

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

U.S. Army Construction Engineering Research Laboratories (USACERL)
P.O. Box 9005
Champaign, IL 61826-9005

8. PERFORMING ORGANIZATION REPORT NUMBER

TR 96/82

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Headquarters, U.S. Army Corps of Engineers (HQUSACE)
ATTN: CEMP-CP
20 Massachusetts Avenue, NW.
Washington, DC 20314-1000

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

Copies are available from the National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161.

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This report presents a methodology incorporating object-oriented and rule-based concepts to generate a preliminary construction plan for facility designs to: (1) compare alternative designs from a construction time and cost perspective; (2) animate the schedule to verify the schedule and identify constructibility problems before construction begins; (3) provide a good baseline schedule and cost estimate for the evaluation of contractor bids; and (4) determine the impact on schedule and costs due to modifications during the construction management phase. A prototype implemented in a LISP-based environment includes: (1) object-oriented models of entities relevant to construction planning; (2) knowledge-bases to generate activities, identify construction methods, and sequence activities; (3) interactive capability to control the level of detail in schedule by grouping components based on spatial location by construction zones; (4) interface with MCACES (Micro-Computer Aided Cost Engineering Support system) databases to obtain crew definitions, unit costs and productivity information for common construction activities; (5) interface with Microsoft Project to do Critical Path Method analysis and display and manipulate schedule information; (6) capability to animate any portion of the schedule; and (7) capability to deal with incomplete design information by using construction solutions from previous projects. The prototype system was well received by schedulers and estimators.

14. SUBJECT TERMS

object-oriented programming
construction scheduling
cost estimating
computer aided design

15. NUMBER OF PAGES

56

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	SAR

Foreword

This study was conducted for Headquarters, U.S. Army Corps of Engineers, Directorate of Military Programs, under Project 4A162784AT41, "Military Facilities Engineering Technology"; Work Unit FF-AS4, "Construction CADD." The technical monitor was Walt Norko, CEMP-CP. The research also was supported, in part, by an appointment to the Research Participation Program at the U.S. Army Construction Engineering Research Laboratories (USACERL), administered by the Oak Ridge Institute for Science and Education through an interagency agreement between the U.S. Department of Energy and USACERL.

The work was performed by the Engineering Processes Division (PL-E) of the Planning and Management Laboratory (PL), USACERL. The USACERL principal investigator was Annette L. Stumpf. Appreciation goes to Dr. Francois Grobler for his insightful comments on several aspects of this work. E. William East also provided useful feedback on early versions of this manuscript. Thanks go also to Jeffrey Heckel, Kirk McGraw, Dr. Michael Case, and Eric Griffith (CECER-PL-E) for providing some of the computer code used in the implementation described in this report. Dr. Case is acting Chief, CECER-PL-E; Michael L. Golish is Operations Chief, CECER-PL; and Dr. David M. Joncich is Chief, CECER-PL. The USACERL technical editor was Linda L. Wheatley, Technical Resources Center.

COL James T. Scott is Commander of USACERL, and Dr. Michael J. O'Connor is Director.

Contents

List of Figures and Tables

Figures

Tables

1 Introduction

Background

The Department of Defense (DOD) spends $1 billion a year to build facilities and another $1.5 billion to operate, maintain, and repair them. At these funding levels, substantial savings could be realized by reducing the number of errors and improving tradeoffs between competing design goals. Many design errors go undetected until the facility is under construction or in operation when it is more expensive to correct these errors. Also, the various disciplines involved in facility delivery tend to work independently because current methods of sharing design information make it difficult for them to share and revise the same design files concurrently. Design participants are therefore unable to see how their individual decisions impact building systems that other participants are responsible for, thus resulting in a suboptimized facility.

This research effort is part of the collaborative engineering program at the U.S. Army Construction Engineering Research Laboratories (USACERL) (Golish 1993). The goals of this program are to improve the efficiency and effectiveness of the facility delivery process through the use of intelligent tools that support a collaborative decisionmaking process. This improvement is achieved within a paradigm known as "agent-based collaborative design" (Figure 1). An agent is a tool that provides support and expertise required for a particular task associated with disciplines such as design, construction, and operations and maintenance.

The Agent Collaboration Environment (ACE) is a prototype system developed by USACERL to provide a platform for agent development, collaboration, and conflict identification (*ACE 1.1 Developer's Manual*, 1995). The construction planning capability described in this report functions as an agent in the collaborative design environment to identify and correct problems before actual construction begins.

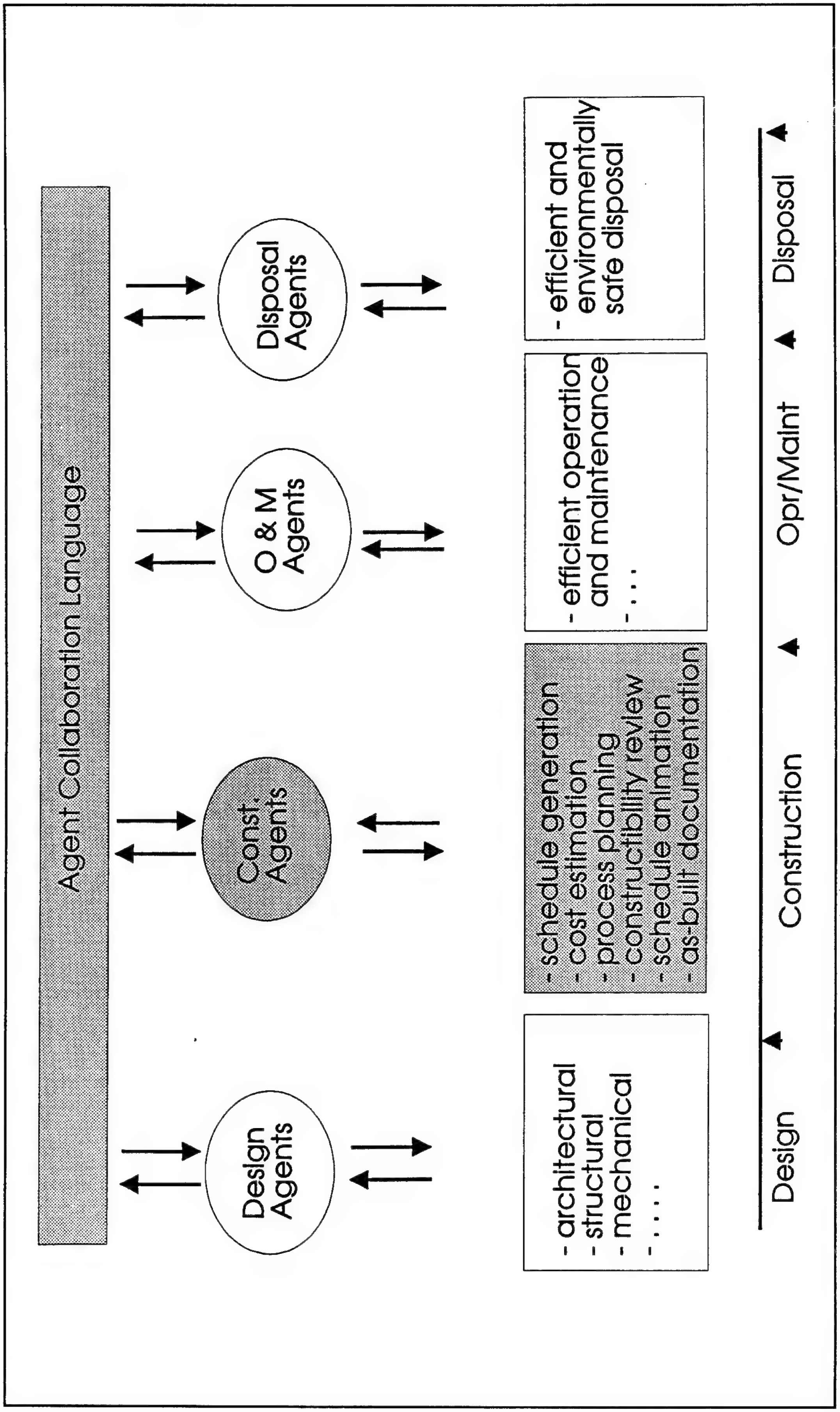


Figure 1. Agent-based collaboration of facility delivery.

Objective

The objective of this research was to develop a methodology incorporating object-oriented and rule-based concepts to generate a preliminary construction plan (or schedule) for facility designs to:

1. compare alternative designs from a construction time and cost perspective
2. animate the schedule to verify the schedule as well as identify constructibility problems and correct them before actual construction begins (design components are linked to the schedule activities by the schedule generation process, so no additional work needs to be done to link the schedule with the design components)
3. provide a good baseline schedule and cost estimate for the evaluation of contractor bids
4. determine the impact on schedule and costs due to change orders and modifications during the construction management phase.

The objective will be achieved by leveraging object-oriented models of the construction planning domain and rule-based technology to capture knowledge and experience used by construction planners.

Approach

USACERL researchers defined issues to be considered in generating a construction plan, reviewed previous efforts related to automated construction planning, and determined the approach to be used in this effort. Implementation of the prototype construction plan application is discussed in Chapter 5.

Mode of Technology Transfer

Two transfer mechanisms are appropriate for various components of research.

Software and knowledge bases

Because of the heavy use of A/Es in military design, this technology must be commercialized through a vendor of computer-aided design (CAD). A Cooperative Research and Development Agreement (CRaDA) would be the mechanism to transfer both the software development environment and applications/knowledge basees developed

using this environment. This approach will share costs with the private sector and reduce technology transfer and maintenance costs.

Software standards for agent collaboration

A collaborative research initiative between several major universities is planned to define an agent collaboration language (ACL) to support interaction between various agent systems. The result of this work will be a proposed national and international standard to the Product Data Exchange System using the Standard for the Exchange of Product Model Data (PDES/STEP) organization represented by the National Institute of Science and Technology (NIST) in the United States.

The Construction Planning Agent was developed using ACE and other software technologies. The support and maintenance for ACE 1.1 are provided by the Engineering Processes Division of USACERL, and the system is available to the DOD community through this agency. The ACE 1.1 User's and Developer's manuals are available through USACERL by calling (217) 325-6511, ext. 6382 or 7511, by calling toll-free at 800-USA-CERL, or writing USACERL, ATTN: CECER-PL-E, P.O. Box 9005, Champaign, IL 61826-9005.

2 Construction Planning

Five important considerations in construction planning appear below in random sequence:

- Determine overall construction strategy
- Decide the level of detail
- Define construction activities (development of Work Breakdown Structure)
- Determine the construction method, resource requirements, and duration for activities
- Determine the sequence for activities.

These issues are interrelated (i.e., crew sizes and the sequence of activities must be considered in defining construction activities). Thus, construction planning is an iterative process where plans may be generated and evaluated a few times before a satisfactory plan is achieved. This research effort is intended to develop an environment to facilitate this process.

Once a preliminary schedule has been developed, the construction planner analyzes and changes it as appropriate (i.e., increasing crew size and thus reducing durations). Thus, the development of a schedule is an iterative process.

Determine the Overall Construction Strategy

One of the first considerations in the construction planning process is to determine the applicability of cost-effective construction strategies such as prefabrication, preassembly, modularization, and other special construction techniques. Prefabrication and preassembly involve the manufacture and assembly of some portion of the building off-site, so that only the final assembly is done on-site. In the case of modularization, a building is divided into fairly complete units (including finishes), which are manufactured off-site and assembled on-site. A report by Tatum et al. (1986) develops guidelines for the effective use of such techniques on building projects.

Decide the Level of Detail

The level of detail in a construction schedule is determined by the intended use for it. The three levels most often identified (Halpin 1976) are: organizational, project, and process. The organizational level involves only key project activities that need to be monitored for timely completion of the project. The project level is more detailed and may identify activities such as excavate foundations, pour concrete for foundations, etc. The process level contains even more detail (i.e., a field schedule), including activities, for example, for excavating a foundation, such as excavate, load, and haul. Sometimes construction contract requirements specify the level of detail by either specifying the minimum number of activities or the maximum duration of an activity (Beach 1993).

Work Breakdown Structure Development

A common method used to identify activities is the development of a "Work Breakdown Structure (WBS)." The WBS is a tree of activities—the activities at the leaf level are the activities that involve the installation of various components. An example of a WBS is shown in Figure 2. The WBS may be developed in a top-down manner, a bottom-up manner, or a combination of both. The important considerations in the process of developing a WBS are: (1) identifying the kinds of activities required to construct the facility, (2) determining the level of detail at which to consider these activities, (3) determining how to spatially combine components to form realistic construction activities, and (4) determining how to aggregate or summarize the activities to create the WBS. Kim's Ph.D. thesis (1993) identifies five factors used to determine WBS for petrochemical plants: (1) block—spatial zone, (2) system—functional unit of the plant, (3) fabrication method, (4) material type, and (5) size.

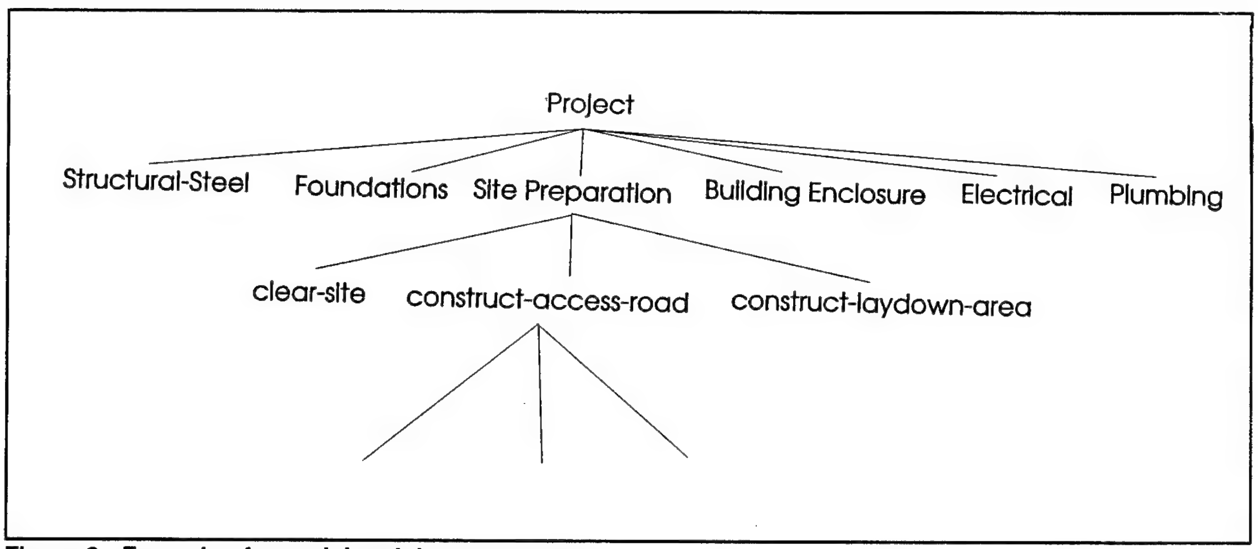


Figure 2. Example of a work breakdown structure.

Identifying the types of activities included in the schedule is determined by the construction activities required to install the design components in the facility. For example, the following activities may be associated with a continuous cast-in-place concrete footing: trench excavation, trim excavation bottom and sides, compact soil, place formwork, place reinforcement, pour concrete, finish concrete, cure concrete, and backfill. Based on soil conditions and other site-specific considerations, one or more of these activities may not be required. For example, the “compact” activity may not be required, or an activity may need to be added to “borrow fill” for backfill if the excavated soil is not suitable for backfilling. The schedule must also include nonconstruction activities such as obtaining permits, material procurement, owner inspection, submittals, etc. (Beach 1993). Some of these activities can be elaborated into more elemental activities depending on the level of detail desired in the schedule. For example, trench excavation involves: (1) excavate and load, and (2) haul load. This research, however, deals only with project level activities.

For the purposes of scheduling, the construction site is divided into areas. Figure 3 shows an example where the scheduler may create two activities (sometimes referred to as work packages): pour concrete for foundations in Area 1 and pour concrete for foundations in Area 2. A number of factors may be involved in defining these “areas.” The areas may reflect natural physical zones in a building such as a floor or part of a floor. The availability of resources for a particular trade may determine the sizing of these areas. Areas also may be constructed on the basis of the amount of work that may be accomplished in 1 working day by the chosen crew size for the particular trade.

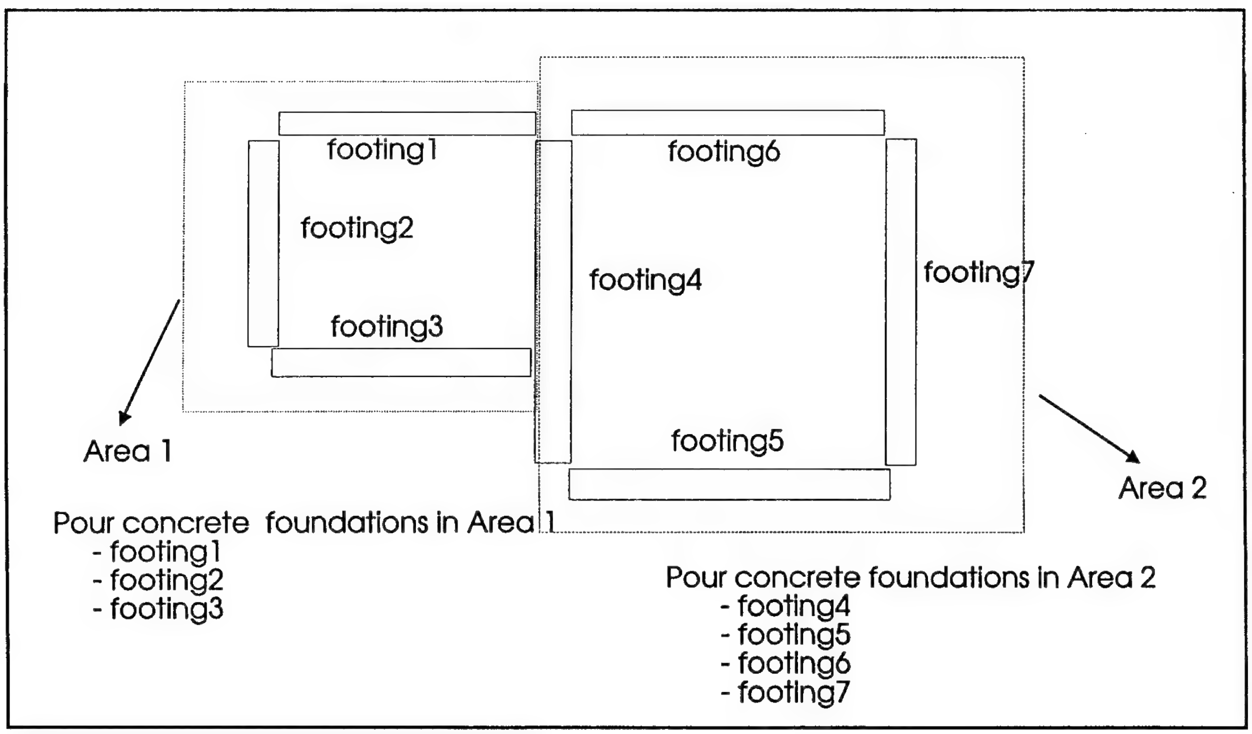


Figure 3. Example of scheduling areas/zones.

Determine Construction Methods, Resource Requirements, and Durations for Activities

Construction Methods

Typical construction activities may be accomplished by different methods. For example, concrete may be poured by using a pump or using a crane and bucket. The choice of a particular method depends on factors such as site accessibility for construction equipment, budget limitations, etc. The choice of construction method also determines the crew requirements for construction activities and may affect the definition and sequencing of activities as discussed below.

Resource Requirements

Computing work quantities requires knowledge of design component dimensions and sometimes other site-specific information such as the soil's angle of repose for computing excavation volumes for foundations. The formulas for computing quantities may be associated with construction activities. Historical cost databases such as the MEANS (R.S. Means Co. 1993) contain information about crew requirements and productivity for many types of construction activities.

Activity Durations

To determine activity durations, the following information is needed: (1) quantity of work from the design component descriptions and other considerations (e.g., soil conditions), (2) number of crews to allocate for the activity, and (3) productivity of the crew. The problem of determining how many crews to use to complete a particular activity is a complicated process involving time and cost tradeoffs. Sometimes, heuristics based on quantities of work are used to determine recommended activity durations. For example, it should take X days to pour Y cubic yards of concrete. These durations are used to determine the number of crews based on the productivity of a single crew. Information from the MEANS can be used to estimate activity durations (assuming a single crew size). However, it must be noted that productivity information found in places like the MEANS are only approximations. Thomas and Smith (1990) cite the following as primary causes affecting productivity: weather, poor sequencing, interruptions, congestion, rework, and restricted access.

Determine Activity Sequencing

Activities in a schedule must usually be performed in a sequence determined by factors such as the physical relationships among components in the design and interactions between crews required to complete an activity. Different types of precedence relationships involve varying degrees of overlap among the activities.

The problem of determining an activity sequence may be considered in two parts: (1) activity sequence involved in the installation of a single component type and (2) activity sequence arising from the spatial and functional relationships between components. In the first case, the activity sequence does not change from project to project. For example, consider the sequence involved in construction of a cast-in-place concrete element: formwork, place reinforcing, pour concrete, finish, and cure. In the second case, the activity sequence varies from project to project based on the physical interrelationships between design components. For example, if floor-1 is supported by beam-1, then floor-1 cannot be installed until beam-1 is installed.

3 Research Efforts Related to Automated Construction Planning

This chapter reviews prior efforts at automating the construction planning process. Because time-cost tradeoffs were not explicitly considered in these efforts, the generation of the cost estimate is a byproduct of the schedule generation process. Thus, the following discussion focuses on schedule generation. In the context of automated schedule generation, in addition to the issues identified in the previous chapter, two additional issues need to be considered: (1) representation and use of building components and systems, and (2) planning architecture (i.e., the codification of schedule generation knowledge within a computer system).

The following research efforts in this area were examed to determine how each addresses the issues outlined above.

1. Construction PLANEX (Zozoya-Gorostiza et al. 1989)
2. OARPLAN (Object, Action, and Resource Plan) (Darwiche et al. 1989)
3. CASCH (Computer-Assisted Scheduling) (Echeverry 1990)
4. KNOW-PLAN (Ayman 1991)
5. Integrating Construction Scheduling With Cost Estimating (Yau 1992).

Construction PLANEX

Representation and Use of Building Components

Construction PLANEX uses a building's description in terms of its low-level components such as concrete column footings, steel beams, steel columns, etc. Each component includes geometric information (x, y, and z dimensions) and additional attributes (e.g., percentage of steel for a concrete footing).

Determination of Construction Methods

Construction PLANEX uses heuristic rules based on type of activity and quantity of work to be performed to identify the construction method and crew to be used for the activity.

Activity Identification

Construction PLANEX distinguishes between element activities and project activities. Element activities are associated with each design component. Project activities represent aggregations of the element activities.

Element activity knowledge sources describe the set of activities required to construct a design element. These knowledge sources contain rules that determine the activities required to construct the design element based on element attributes and site conditions.

The element activities generated by these knowledge sources are then aggregated into project activities. For example, formwork of column footings are aggregated into a project activity that groups all formwork of foundation elements. Similarly, formwork of columns are aggregated into a project activity that groups all formwork for the columns on a particular floor.

Activity Duration Estimation and Crew Allocation

For each type of project activity, knowledge sources (a collection of knowledge generally encoded as if-then rules) are used to determine the recommended duration of the activity. One rule used in knowledge sources might be: "If the quantity of work is less than 6,400 units, the recommended duration is 5 days."

These recommended durations are used to estimate the crew sizes for each project activity. The crew sizes are then used to compute real durations.

Activity Sequencing

Knowledge sources are used to generate successors for project activities. For example, the sequencing knowledge for project activity "concrete pouring for column footings" is expressed in the form of two rules that identify the successors: "form stripping for column footings" and "form placement for columns on footings." These rules do not use any knowledge of relationships among the design elements.

Planning Architecture

The architecture used for process planning by Construction PLANEX has four main components: representational structures, operators, knowledge sources, and a user interface. Darwiche et al. (1989) contains details about these components. The operators in Construction PLANEX are procedural functions that modify objects in the

context (global data store). Construction PLANEX can generate a plan either interactively or in a fully automated manner. In the fully automated mode, the following operations are executed (in the sequence indicated):

1. Build the tree of design elements from a description of the facility
2. Generate the set of element activities to construct each element
3. Link the element activities into a tree
4. Compute the amount of work for each element activity
5. Determine the unit of measure for the amount of work for each element activity
6. Select the material package used by each element activity
7. Synthesize the project activities from aggregated element activities
8. Link the project activities into a tree
9. Select the technology for each project activity
10. Compute quantity of work for each project activity (sum of the quantities of work for each element activity that is aggregated to form the project activity)
11. Determine recommended durations for each project activity
12. Determine how many crews to allocated to each project activity
13. Determine duration for element activities
14. Establish precedences, leads, and lags among project activities
15. Compute estimated cost for each project activity
16. Apply CPM algorithm to schedule the project.

Some of these operators are purely algorithmic (e.g., CPM algorithm) while others may involve the evaluation of rules.

OARPLAN

Representation and Use of Building Components

OARPLAN requires information about the type of building component and the relationships among the components. The file for describing facility components has the following format:

<component-type> <component-id> <relationship> <related-component-ids>

For example, (slabs s0 supported-by (F1 F2 F3)).

Determination of Construction Methods

OARPLAN in its current version does not consider alternative construction methods for construction activities.

Activity Identification

The identification of activities in the schedule occurs in two ways: (1) activity scale reduction and (2) activity subplans. For activity scale reduction, for example, construct building-1 is elaborated into construct floor-1, construct floor-2, and construct floor-3. For activity subplans, construct concrete element is elaborated into (1) pour concrete, (2) finish concrete, and (3) cure concrete.

Activity Duration Estimation and Crew Allocation

Presently, OARPLAN has no facilities for automatically determining the duration of activities. The estimates for duration are determined manually using historical data sources such as the MEANS Building Construction Cost Data. An action in OARPLAN is associated with a list of possible resources. However, in the current version, the allocation of resources is not based on construction methods or component attributes.

Activity Sequencing

The dependencies among activities are expressed in two ways: (1) as part of activity subplans (e.g., pour concrete precedes finish concrete) and (2) in the form of rules in the following form:

- If activity-1 and activity-2 are in the plan
  and activity-1 is linked to object-1
  and activity-2 is linked to object-2
  and object-1 is related to object-2 by the dependency relationship P
  then activity-2 is a predecessor of activity-1.

Relationships are designated: supported-by, enclosed-by, connected-to, covered-by, weather-protected-by, and damaged-by.

Planning Architecture

The initial version of OARPLAN was implemented using a multiple blackboard-based environment. It was organized as four blackboards, each having a particular function: the facility blackboard, the action blackboard, the plan blackboard, and the elaboration

and dependency knowledge sources blackboard. The knowledge sources whose preconditions are satisfied post rules in a "triggered agenda." Those rules, whose preconditions are satisfied are transferred to an "executable agenda." A score is computed for each rule based on the current control strategy and the rule with the highest score is executed causing changes to the blackboards. This cycle is repeated. OARPLAN does not use sophisticated control strategies.

CASCH

Representation and Use of Building Components

CASCH uses a predefined building system breakdown based on the Building Systems Index (BSI) format. The building definition for a particular project is input by the user selecting from alternative subsystem options (e.g., exterior walls are masonry or precast). Relationships among components are predefined in this representation.

Determination of Construction Methods

Construction methods are not considered for activities.

Activity Identification

Activity breakdown is represented by three levels of detail. The first level activity is Building Construction. The eight activities in the second level include: Site Preparation and Foundation Work, Frame Erection, Rough-in Work, Roof Work, Skin Insulation, Floor Finishing, Elevator Work, and Site Finishing. The third level contains the detailed activities required to install and remove various building components.

Activity Duration Estimation and Crew Allocation

Approximate rules are used to determine the duration of activities. Approximate quantities are derived based on gross dimensions of the building, which are then used to compute durations based on productivity data found in MEANS. Crews are not allocated to activities.

Activity Sequencing

Activity sequencing occurs based on four factors:

1. physical relationships among building components (e.g., supported-by, covered-by, embedded-in, and requirement of service)
2. interaction among crews, equipment, and materials
3. requirement of an interference-free path for components and their installation
4. code regulations that ensure the safety of construction operations and the ability to supervise and inspect installed components.

An example of a sequencing rule is: If component-X is supported by component-Y, then activity to install component-Y precedes activity to install component-X.

Planning Architecture

CASCH is an interactive system in the KEE®* environment. The system is organized in four knowledge modules: (1) Building Systems, (2) Activity Identification, (3) Duration Estimation, and (4) Activity Sequencing. For a particular scheduling run, the user interactively defines the building instance and invokes the activity identification, sequencing, and duration estimation operations.

KNOW-PLAN

The main objective of this research was to develop methods to use geometric data to provide a dynamic sequencer for project planning.

Representation and Use of Building Components

A three-dimensional geometric model of facility components is generated. The geometric data associated with each object includes: minimum x,y,z coordinates, maximum x,y,z coordinates, and the center and rotation in x,y,z. Each component is also associated with a class that specifies the direction of installation. Each object is assigned an attribute indicating the type of connection (i.e., structurally supported, embedded-in, protected-by, etc.) with other objects around it.

---

*Intellicorp, Inc., 1975 El Camino Real West, Mountain View, CA 94040-2216.

Activity Identification

This system has no facility for identifying activities automatically. Activities are provided as input. Also, each object in the computer model is associated with one activity in the construction plan.

Activity Duration Estimation and Crew Allocation

No facility exists for automatically computing activity durations. Schedule attributes such as duration, early finish, etc. are input by the user.

Activity Sequencing

The geometric reasoning process asserts the sequence between two objects based both on the geometric relationships between the objects and the relationships between the classes to which the objects are related. If objects belong to the same class, then direction of installation is used to assert the sequence. When objects belong to different classes, the connection type information and the geometric information are used to assert precedences.

Sequence is determined by activity networks such as: (1) a geometric network based on spatial relationships, connection types, and classes to which objects belong, (2) a constructibility network derived by using the pathfinder routine of a Walkthru™* system to simulate installation of objects, and (3) other networks such as resource constraints, mandatory dates, and procurement constraints, which may be defined by the users. The network links have priority values that are used to resolve conflicts.

Integrating Schedule Generation and Cost Estimation

Yau's work (1992) extends CASCH to include information about costs to generate a cost estimate along with the schedule. The cost estimate and schedule are integrated by introducing the concept of a "task," which is defined as: "the quantity of work performed by a single crew for the installation or preparing of the installation of one or a group of similar design components" (Yau 1992). Each task has alternative methods for performing the task. The task methods are related to crew and material cost items. Yau gave each design component a set of related tasks that are required to install the component. The design component hierarchy is the same as the one used in CASCH (based on the BSI). Activity durations and precedences are predefined for midrise buildings.

---

*Bechtel Corporation, 50 Beale Street, San Francisco, CA 94119-3965.

Review Conclusions

While Construction PLANEX is fairly comprehensive in that it considers construction technologies, crew allocations, and activity durations, the approach to sequencing activities (based on relationships between objects) used in OARPLAN and CASCH is more attractive. Two major areas that these approaches lack are: (1) flexible definition of "realistic" construction activities (i.e., WBS) and (2) consideration of spatial requirements associated with construction activities.

An evaluation of OARPLAN in the context of construction planning for a "real life" application (Winstanley et al. 1993) determined that the component-level plan was too detailed and complex to be useful. Component-level activities needed to be aggregated into "realistic" work packages. The solution developed was called zoning, which aggregates component-level activities into zone activities based on the assignment of components to zones (Winstanley et al. 1992).

Thomas and Smith (1990) have shown that construction productivity is affected by site congestion and restricted access. Hence, it is important to model spatial requirements associated with various activities so that they can be considered in the selection of construction methods, productivity and duration estimation, and sequencing. This requirement is not handled in Construction PLANEX or OARPLAN. CASCH allows a slot for work area and location but does not allow for the definition or consideration of relationships among these areas.

4 Approach to the Construction Planning Process

Chapters 2 and 3 have described the important considerations in construction planning and discussed the extent to which prior research efforts have addressed these issues. As indicated earlier, the various considerations in the planning problem are interdependent. For example, to determine the degree of site congestion, the activities have to be scheduled. But the productivity of crews (and hence the duration of activities in the schedule) is affected by the degree of congestion on the site. Such interdependencies can be resolved only by an iterative approach. Figure 4 shows the construction planning process as it is envisioned in this approach. The goal is to build a complete construction planning environment with facilities for:

- defining flexible work breakdown structures based on spatial zones, system or component parameters, and other considerations

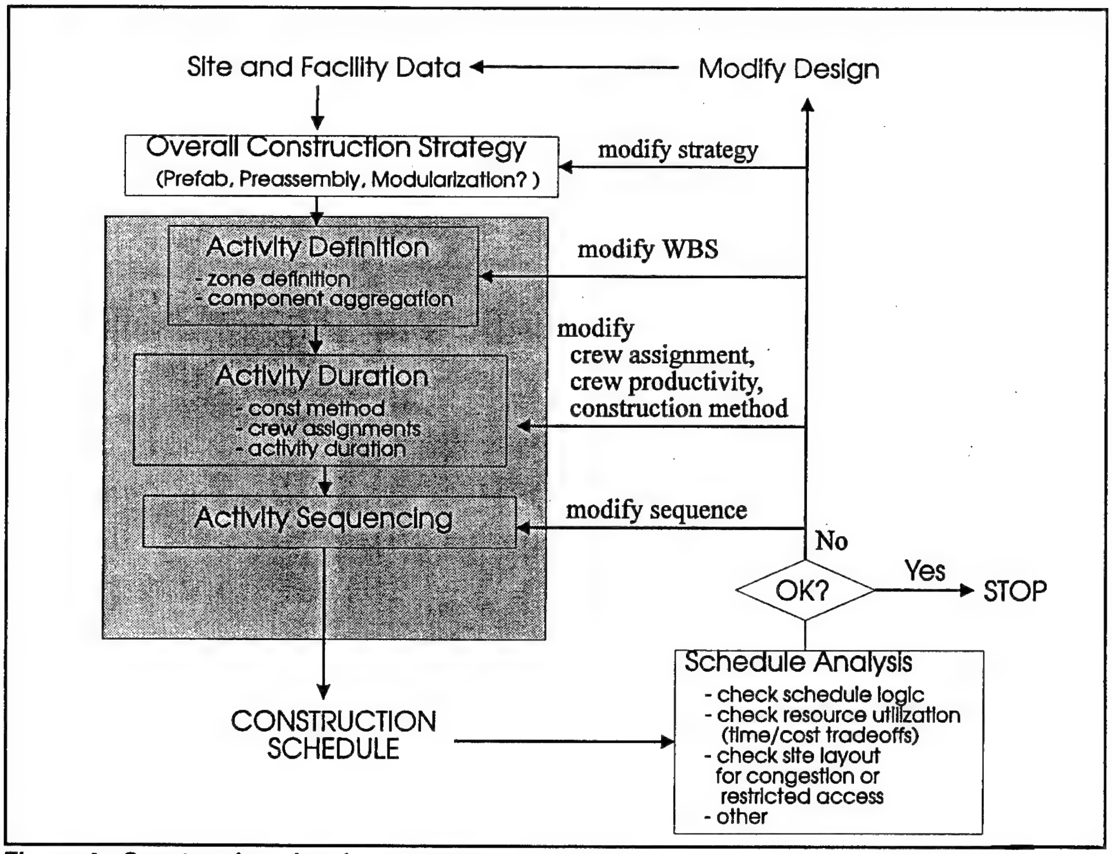


Figure 4. Construction planning process.

- generating activities required for the installation of building systems and components, choosing alternative construction methods for construction activities
- assigning appropriate resources (with defaults provided from sources such as Micro-Computer Aided Cost Engineering Support [MCACES] system developed by USACE as a detailed bottom-up cost estimating tool)
- locating unit cost and productivity information in sources like MCACES
- animating the construction schedule. The scope of automation considered in this effort is indicated by the shaded area in Figure 4. The process of generating a preliminary schedule will be automated. However, the analysis and modification of the preliminary schedule (or any design changes) will be performed by the user.

The conceptual approach used for schedule generation is different from traditional artificial intelligence (AI) approaches to solving planning problems (Fikes 1971; Sacerdoti 1977). It is similar to the OARPLAN approach called model-based planning. However, this approach differs from OARPLAN in that:

- The goal is to develop a complete construction planning environment that considers all aspects of the planning process. (In OARPLAN, the focus is the generation of schedule sequence based on intercomponent relationships; construction methods, crews, quantities of work, and durations are not considered.)
- This is envisaged to be an interactive environment where every decision made by knowledge base can be overridden interactively by the user.
- This approach aggregates components into a "component group" based on spatial (i.e., zone) and similarity considerations, and activities are generated to install these component groups, whereas in OARPLAN, activities were generated for every single component and then aggregated by zones.
- OARPLAN is implemented using a blackboard-type of architecture (Nii ...), whereas this approach uses a task-specific architecture.

The following sections elaborate on important aspects of this approach including: (1) the object-oriented modeling of the types of objects necessary for construction planning, (2) the representation of construction planning knowledge for the definition of construction activities, computing activity durations, sequencing activities, and estimating construction cost, and (3) capabilities for dealing with incomplete design information.

Object-Oriented Modeling for Construction Planning

An object-oriented approach is used to model the entities (i.e., activity, component, resource) in the construction planning domain (Figure 5). The symbols and notation (based on Object Modeling Technique notation, Rumbaugh et al. 1991) in Figure 5 are used to express all the object diagrams in this report.

- The *Component* class represents the physical aspect of a building such as the floor, column, beam, wall, foundation, and many other building components. *Component* objects can have other components for parts as represented by the part-of relationship.
- The *Shape* class (including those classes derived from it) are used to model the geometry of building components (Heckel 1995).
- The *Activity* class represents the construction process information. Many activities are at a task level in a typical facility construction, so the aggregation relationship "is-part-of" is used to represent hammock activities. Precedence relationships among activities are represented by the "precede" relationship.
- The *Schedule* class is an aggregation of *Activity* objects.
- The *Construction Method* class encapsulates information about construction methods for activities like excavation and placing concrete.
- *Resource* objects may be labor, equipment, and crew (which are a specific combination of labor and equipment items).

This approach assumes that building information is in the form of objects. Although a typical CAD environment shows building information graphically in a drawing (which cannot be accepted by the construction planning agent), efforts are under way to develop standard object-based representations for building-related information. When successfully completed, these programs will make translation possible because the formats of object models need not be the same.

As discussed in the previous chapter (page 23), a number of factors need to be considered in the identification of scheduling zones for specific construction activities. Scheduling zones group components that will be scheduled as a single activity (i.e., all the doors of a facility would constitute a zone). The definition of these areas often depends on the type and quantity of work to be performed to construct a facility, the crew size and productivity, and the spatial layout of the facility components. A capability to aggregate design components by spatial location is provided because the spatial design component breakdown (which may be generated by the designers) may not be appropriate for the construction schedule.

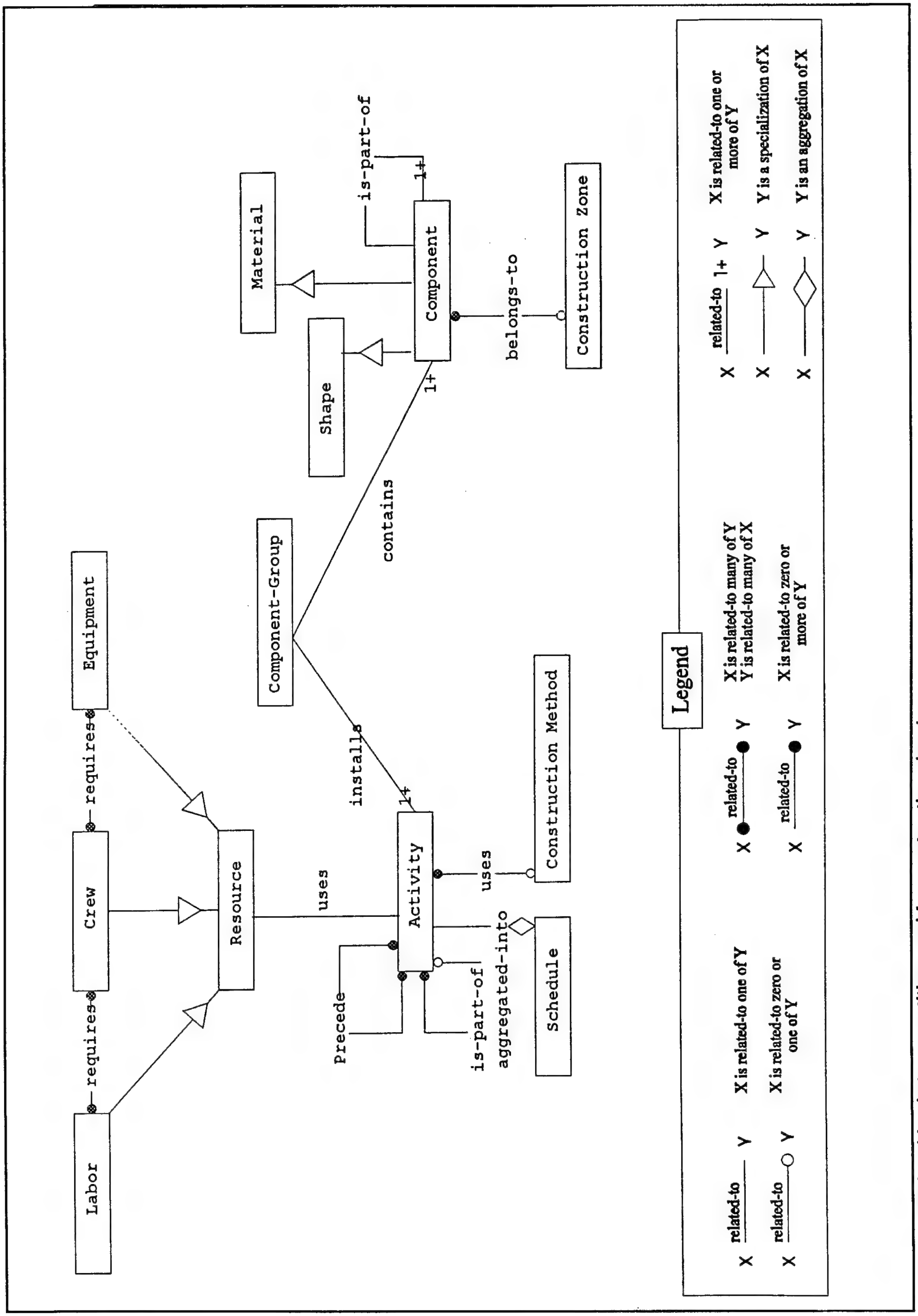


Figure 5. Relationships between entities used for construction planning.

- The *Construction Zone* and *Component Group* classes provide a capability to aggregate design components either by spatial location or by similarity of attributes. This allows the user to control the level of detail in the construction schedule. Activities are generated for the installation of *Component Group* objects.

The following sections discuss some of these classes in more detail.

Building Materials, Components, and Assemblies

Figure 6 shows a portion of the Material class hierarchy used to organize information about building materials. The hierarchy is based on the sfB classification (Jones 1976) though it does not correspond with it on a one-to-one basis. Figure 7 shows a portion of the Building Component class hierarchy, which is a multiple-inheritance hierarchy where Component classes may inherit from higher-level components and from the Material and Shape classes as in the case of continuous footings and standard slab-on-grade. While the Building Component classes hierarchy is based on the Uniformat classification (Uniformat, 1992), it does not necessarily correspond with it on a one-to-one basis. An assembly is a building component comprised of other components that may be assemblies themselves. Figure 8 shows an example of an exterior-wall assembly.

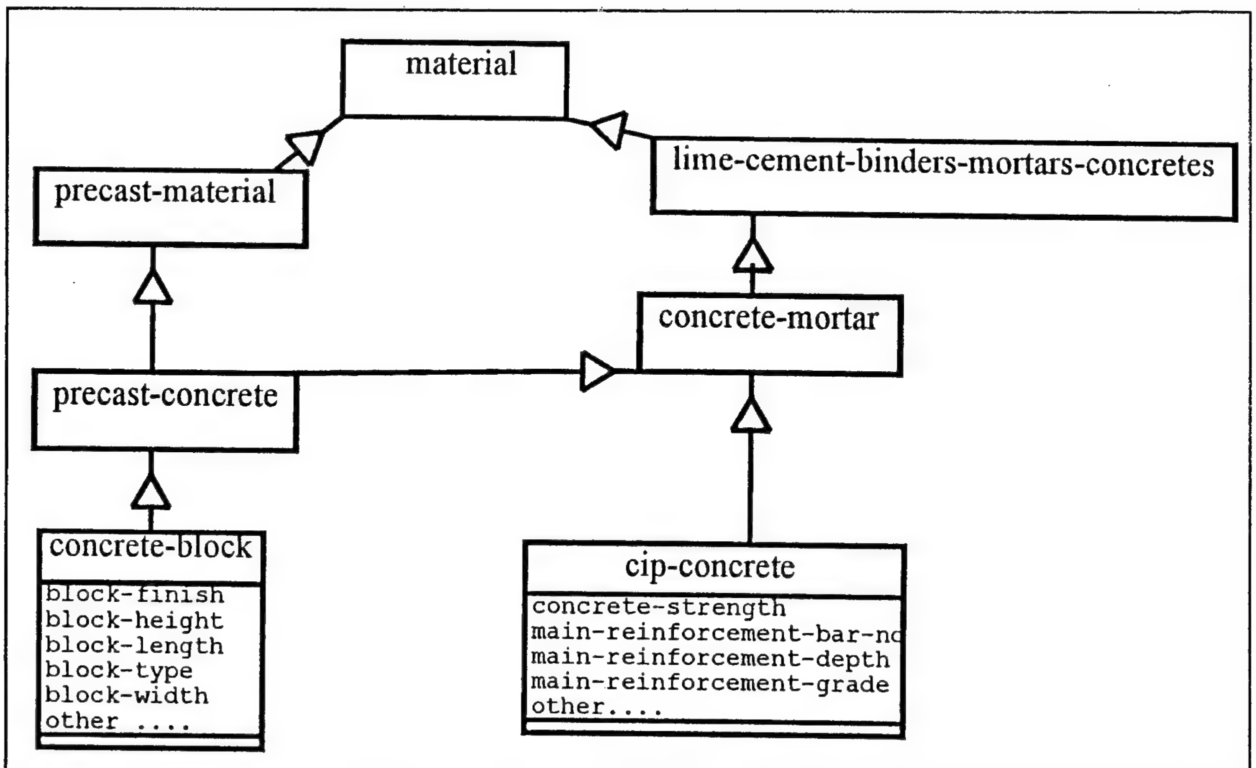


Figure 6. A portion of the material classes hierarchy.

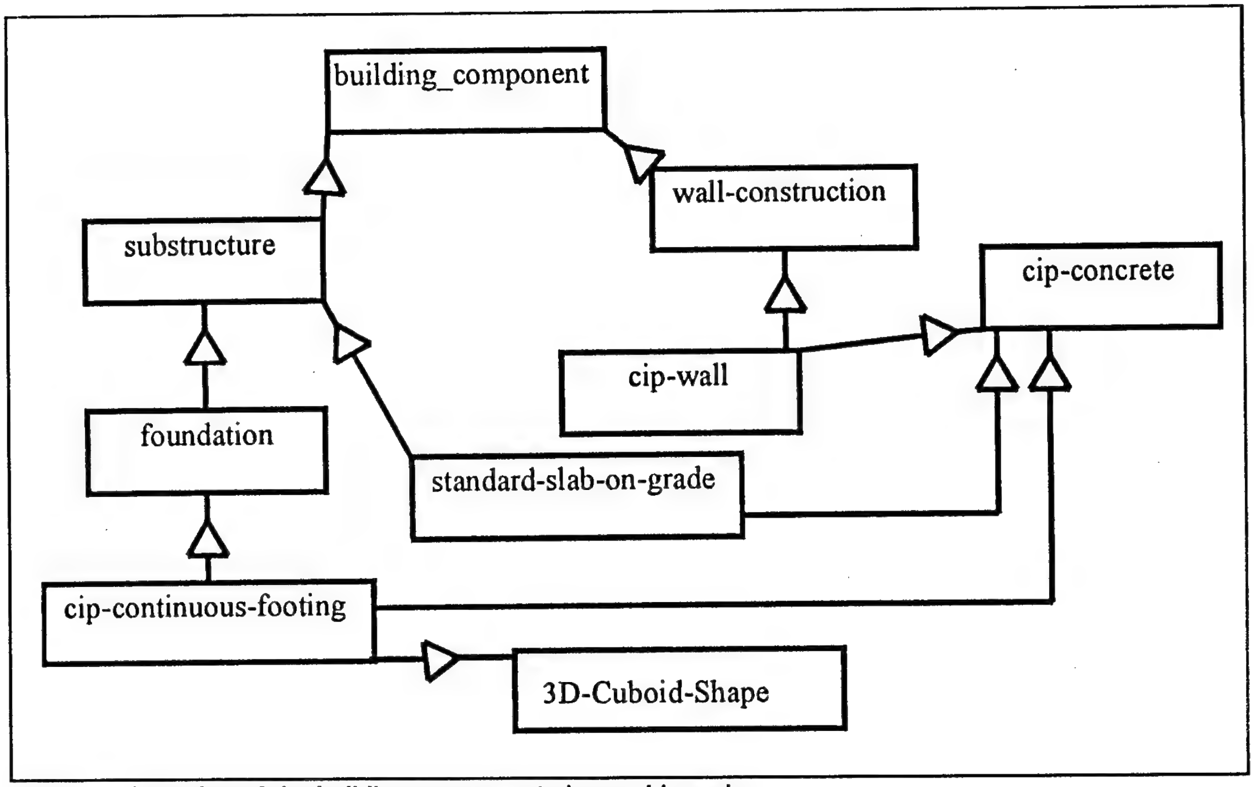


Figure 7. A portion of the building component classes hierarchy.

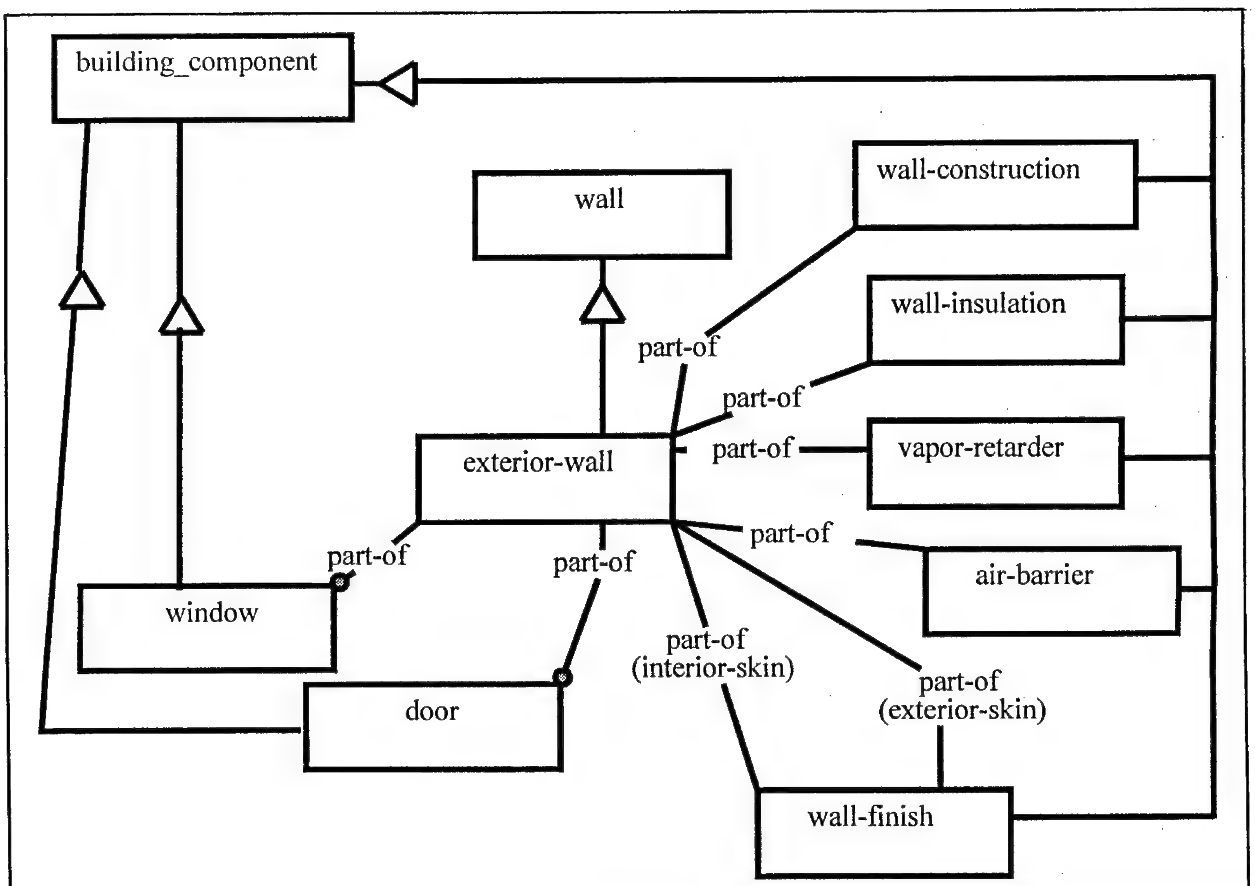


Figure 8. Example of an assembly (exterior-wall).

Construction Activities, Methods, and Resources

The Construction Activity hierarchy organizes the information on construction activities. Activities may be classified into the following major categories: Procure, Deliver, Submit, Install, Approve, Dummy (Nomani et al. 1992). Existing schemes such as the Activity Definition Index (Nomani et al. 1992) or the CSI Masterformat* are used as a starting point for the development of such a hierarchy. Figure 9 shows a portion of the Construction Activity class hierarchy. The top-level class in this framework will be the activity class with a definition that includes attributes common to all activities such as: name, duration, early-start, early-finish, late-start, late-finish, etc. All other classes inherit from the Activity class adding additional information necessary for the specific activity type. For example, the Earthwork class inherits the attributes from the Activity class and defines additional attributes such as: soil-type, soil-moisture-content, etc. Further, the Excavating activity class inherits from Earthwork and defines additional attributes like: angle-of-repose, length-of-excavation, width-of-excavation, depth-of-excavation, bracing-required, dewatering-required, hauling-required. Instances of these activity classes would be instantiated for a particular schedule.

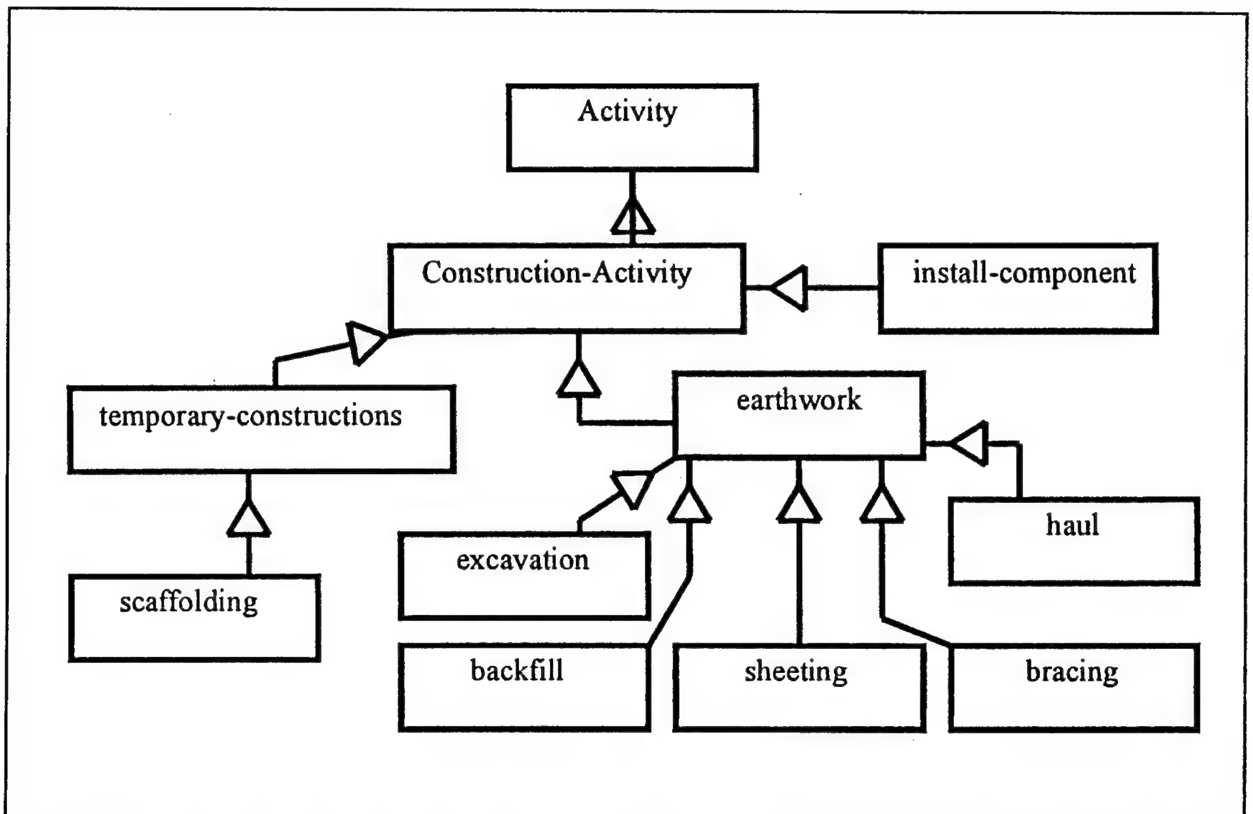


Figure 9. A portion of the Construction Activity classes hierarchy.

*Construction Specifications Institute, Alexandria, VA.

In this work, the labor, equipment, and crew definitions found in the MCACES unit price database was used. Figure 10 shows a portion of the Resource class hierarchy.

Versioning

A versioning capability is important as it allows us to keep track of changes to data allowing for the possibility of reverting to a previous state and also maintaining alternative solutions in a "what-if" scenario. In this approach to data modelling, an object (or instance) is an instance of a class (or frame). Objects have slots that store "nonobject" values (i.e., they can be any programming type except objects). Relationships between objects are explicitly modelled using semantic links. Thus, the versions for an instance can differ in: (1) values for their slots and (2) relationships to other objects. In such a situation, an important consideration is the propagation of versions across object relationships. Consider an object "Wall-1" that is related by a "has-part" relationship to another object "door-1." If a new version of "Wall-1"—"Wall-1-1"—is created, should a new version of "door-1" also be created automatically? Similarly, if a new version of "door-1" is created, should a new version of "Wall-1" be created automatically? The most flexible approach is to design the versioning functionality with options to allow this sort of propagation across relationships. In this approach, versions are treated as instances that are related by a "version derived from" relationship (Figure 11). A status slot is used to store information about which version is current.

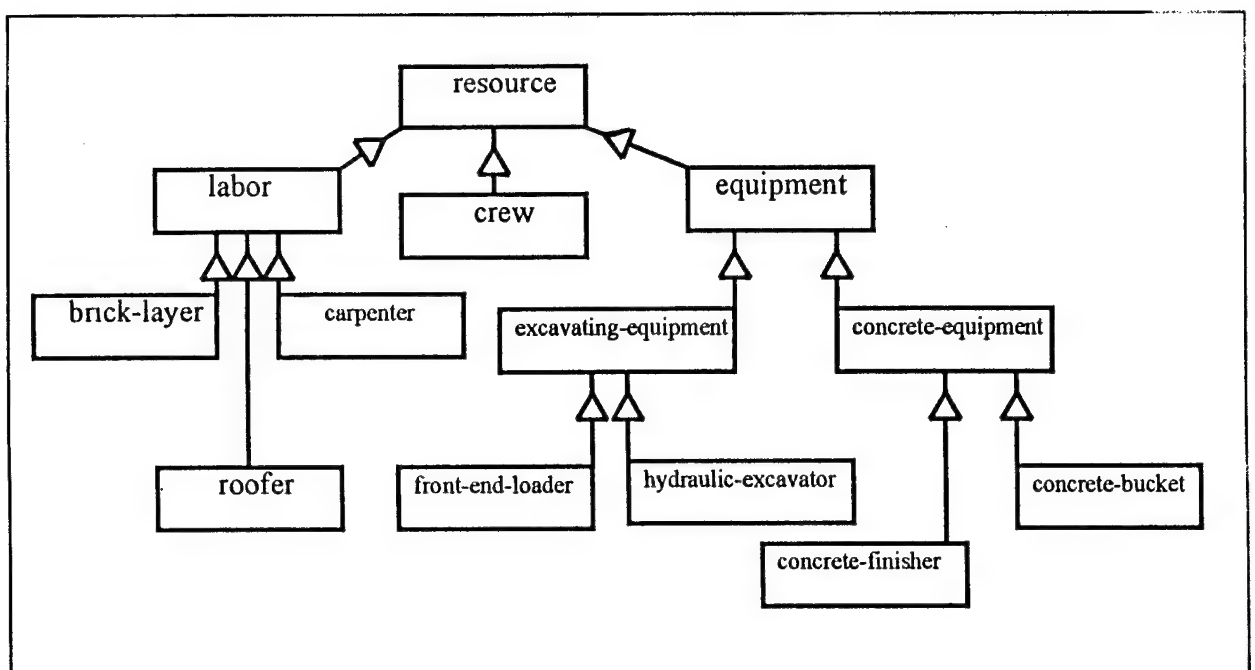


Figure 10. A portion of the Resource classes hierarchy.

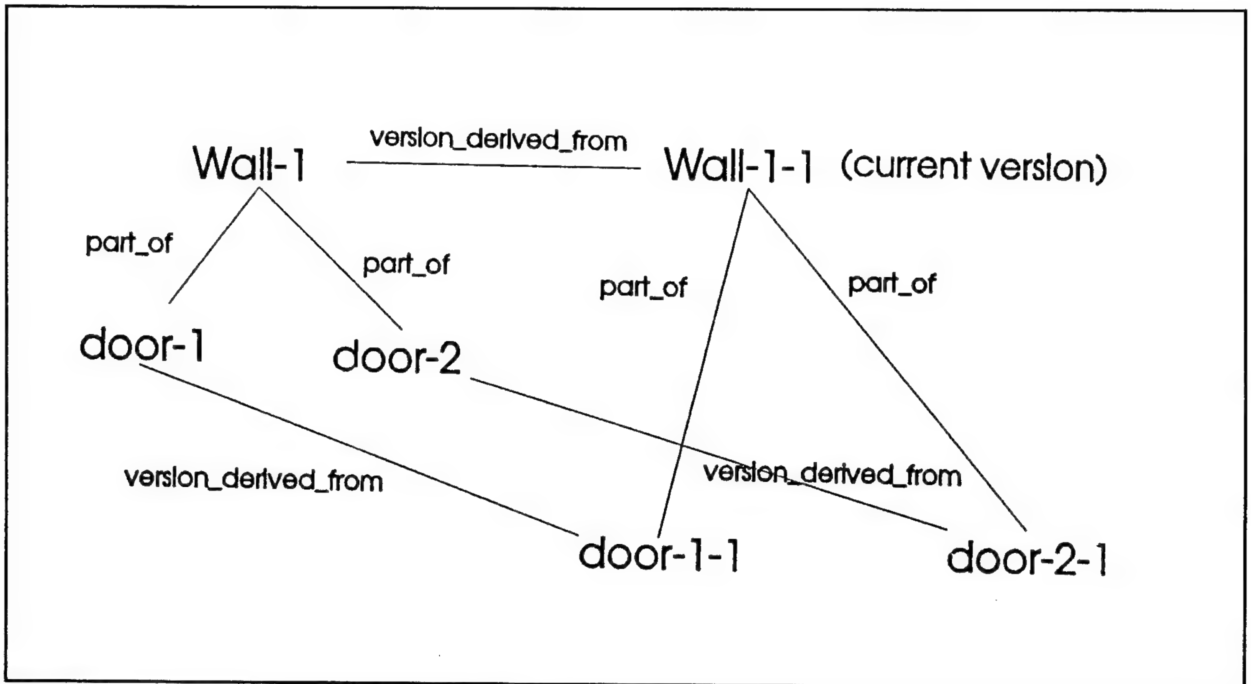


Figure 11. Representing versions of objects.

Construction Planning Knowledge Representation

This approach uses the object-oriented building models described on page 26 to organize knowledge needed for construction planning including: (1) identifying activities required to install design components and assemblies, (2) computing activity parameters and quantities of work, (3) identifying construction methods, (4) assigning crews based on historical project databases, and (5) sequencing activities. The knowledge, developed from textbooks and experienced schedulers, is represented as methods and rules associated with the classes in the object hierarchies described earlier. The rules are grouped together on the basis of the kind of knowledge they express and the particular object class to which the knowledge is related. For example, rules to identify construction activities required for the installation of components may be grouped by material or component class: identify-activities-<material-class> or identify-activities-<component-class>. Such an organization of rules is necessary for maintaining the knowledge base, especially as it increases in size. For the purposes of this research, no effort was made to acquire an extensive rule/knowledge database.

Construction Activities Definition

The process of defining construction activities requires knowledge of the project-level activities required to install each type of component, including supporting activities like scaffolding, procurement, etc. Also, to generate a manageable schedule, the entire

site is divided into construction "zones" and activities are generated for groups of components within these zones. Once the activities have been generated at the detailed level, it is possible to automatically generate work breakdown structures that summarize the detailed activities in various ways.

The activities needed to install the various component types are generated by rules. The rules are organized by: (1) type of material used to construct the building component (e.g., cip-concrete, unit-masonry) and (2) type of component (e.g., standard foundations such as continuous-footings and column-footings require excavation and backfill or subbase preparation for standard slab-on-grade). For example, the following activities are common to all cast-in-place concrete components: erect formwork, place reinforcement, pour concrete, finish concrete, cure concrete, and remove formwork. These activities are codified in the rule shown below:

```
(define-rule cip-concrete-rule
(:direction :forward)
(instance ?query is IDENTIFY-ACTIVITIES-QUERY)
(bind ?cmpgrp (send-msg ?query :component-group))
(instance ?cmpgrp is COMPONENT-GROUP)
(bind ?comp-inst (send-msg ?cmpgrp :standard-component-instance))
(instance ?comp-inst is CIP-CONCRETE)
THEN
    (instance ?query is IDENTIFY-ACTIVITIES-QUERY
    with activities-list
        (ERECT-CONCRETE-FORMWORK
        PLACE-CONCRETE-REINFORCEMENT
        PLACE-CONCRETE
        CURE-CONCRETE
        REMOVE-CONCRETE-FORMWORK))
    (print-out "cip-concrete-rule .. fired")
```

Other activities such as expansion joints are not included because they are not common to all cast-in-place concrete components (only slabs and walls have expansion joints). Such activities are associated with the specific component types. For example, activities for a standard foundation (applies to cip-continuous-footings, column-footings) include: excavation and backfill.

Certain activities may require supporting activities. For example, for the excavation activity, if the soil is saturated (moisture content is wet) then the "dewatering" activity needs to be performed as shown by the following rule:

```
(define-rule excavation-supporting-activities-dewatering-rule
(:direction :forward)
(instance ?query is GENERATE-SUPPORTING-ACTIVITIES-QUERY)
(bind ?act-inst (send-msg ?query :activity))
(bind ?site-info (send-msg ?query :site-info))
(instance ?site-info is SITE-INFO
        with soil-moisture-content wet)
        (instance ?act-inst is EXCAVATION)
        THEN
        (instance ? is GENERATE-SUPPORTING-ACTIVITIES-RESULT
        with supporting-activity (DEWATERING))
        (print-out "excavation-supporting-activities-dewatering-rule .. fired")
```

Computing Activity Parameters and Quantities of Work

Formulas and methods are required to compute activity parameters (e.g., depth and width of excavation) and to compute quantities of work (e.g., volume of excavated soil) associated with each activity. For example, for excavation activity, the depth, width, and slope of excavation need to be computed as shown by the following rule that determines the slope (vertical or natural) for excavation:

```
(define-rule excavation-params-nat-slope-rule
        (:direction :forward)
        (instance ?query is COMPUTE-PARAMETERS-QUERY)
        (bind ?act-inst (send-msg ?query :activity))
        (bind ?site-info (send-msg ?query :site-info))
        (instance ?act-inst is EXCAVATION
            with depth ?depth-ft :unit feet)
        (instance ?site-info is SITE-INFO
            with soil-type ?soil-type
            with soil-moisture-content ?soil-mc
            with soil-firm ?soil-firm)
        (or (equal ?soil-mc 'dry) (equal ?soil-mc 'moist))
        (and (equal ?soil-firm 'no) (< ?depth-ft 12))
        THEN
        (instance ?act-inst is EXCAVATION
            with soil-type ?soil-type
            with bracing-reqd no
            with slope natural)
```

```
    (print-out "excavation-param-nat-slope-rule .. fired")
```

The rule states that if the soil moisture content is "dry" or "moist," or if the soil is not firm but the depth of excavation is less than 12 ft,* then no bracing is required and the natural slope of the soil can be used.

Similarly, rules are used to express knowledge needed to compute the quantities of work associated with each activity. For example, the following rule computes the quantity of work associated with placing reinforcement for a standard continuous footing. The quantity of reinforcement is measured in tons.

```
(define-rule compute-quantities-concrete-reinforcement-cont-ftg-rule
    (:direction :forward)
    (instance ?query is COMPUTE-QUANTITY-QUERY)
    (bind ?act-inst (send-msg ?query :activity))
    (bind ?std-comp-inst (send-msg ?query :standard-component-instance))
    (instance ?act-inst is PLACE-CONCRETE-REINFORCEMENT)
    (instance ?std-comp-inst is CIP-CONTINUOUS-FOOTING
        with width ?w :unit feet
        with height ?h :unit feet)
    (bind ?comp-group (send-msg ?query :component-group))
    (bind ?qty (send-msg ?comp-group :sum :method :length))
    (bind ?takeoff (car ?qty))
    (bind ?takeoff-unit (cadr ?qty))
    (equal ?takeoff-unit 'feet)
    (instance ?std-comp-inst is CIP-CONCRETE
        with main-reinf-percent ?psteel)
    THEN
    (instance ?act-inst is PLACE-CONCRETE-REINFORCEMENT
        with quantity (evaluate (list (list 'reinforcement

            (* ?w ?h ?takeoff ?psteel 13.230 0.45 (/ 1 27))

            'TON))))
    (print-out "compute-quantities-concrete-reinforcement-cont-ftg-rule .. fired")
```

*1 ft = 0.305 m.

Construction Method Identification

The choice of a construction method for an activity depends on a number of factors such as local site conditions, availability of labor and materials, etc. The example below identifies the conditions (the depth of excavation is less than 12 ft, the width is less than 6 ft, and the soil is firm) under which a "wheel-trencher" should be used for excavation:

```
(define-rule excavation-const-methods-wheel-trencher-rule
    (:direction :forward :priority 10)
    (instance ?query is IDENTIFY-CONSTRUCTION-METHOD-QUERY)
    (bind ?act-inst (send-msg ?query :activity))
    (bind ?site-info (send-msg ?query :site-info))
    (instance ?act-inst is EXCAVATION
        with depth ?d :unit feet
        with width ?w :unit feet
        with slope vertical)
    (is-number ?d)
    (is-number ?w)
    (and (<= ?d 12) (<= ?w 6))
    (instance ?site-info is SITE-INFO
        with soil-firm yes)
    THEN
    (instance ?query is IDENTIFY-CONSTRUCTION-METHOD-QUERY
        with const-methods-list (wheel-trencher-cm))
    (print-out "excavation-const-wheel-trencher-rule .. fired")
```

The knowledge required to identify construction methods is hard to obtain and codify, so the process of method selection is expected to involve user interaction. Presently, most rules identify the known construction methods for a particular activity type as shown below for the "place-concrete" activity (which includes "pump-concrete-cm" and "direct-chute-cm") and allow the user to select the appropriate method for the specific site conditions.

```
(define-rule concrete-placement-const-methods-general-rule
    (:direction :forward :priority 0)
    (instance ?query is IDENTIFY-CONSTRUCTION-METHOD-QUERY
        with activity-type ?act-type
        with-unknown const-methods-list ?l)
    (equal ?act-type 'PLACE-CONCRETE)
    THEN
    (instance ?query is IDENTIFY-CONSTRUCTION-METHOD-QUERY
        with const-methods-list (PUMP-CONCRETE-CM DIRECT-CHUTE-CM))
```

(print-out "concrete-placement-const-methods-general-rule .. fired")

An assumption is made that the activity generation process is independent of construction method selection. Construction methods may be specified at various levels of detail corresponding to the activities in the project. It is possible that the choice of construction method can affect both the generation of activities and their sequencing. Future research should consider this possibility, perhaps by incorporating a multilevel activity generation process (i.e., a first level of activities are generated for which construction methods are selected; then a second level of activities is generated based on the choice of the method).

Resource Assignment and Activity Duration Computation

The MCACES unit price database (Building Systems Design, Inc., 1992) contains the standard crew, productivity, and unit cost information for the construction activities involved in the installation of common building components. Table 1 shows a portion of this database. The "CREW" field in the above table contains the crew identifier for a standard crew defined by the MCACES crew database, shown in Table 2.

All prices are later adjusted for location. Although the information in the unit price database is organized according to the CSI Masterformat specification, at the detailed levels the selection of appropriate items must be performed by interpreting the description associated with it (see the "DESC" field of the tables). A subset of these

Table 1. A portion of the MCACES Unit Price database.

XK	BASIC	SFFX	DESC	CREW	Unit Material Cost	Unit	Productivity per hour
MIL	02221	1202	excv with hyd.excv with capacity 0.5 cy and rate of 75 cy/hr	CODEA	0.0	CY	63.5
MIL	02221	1203					
MIL	02221	1204					

Table 2. A portion of the MCACES Crew database.

XK	CREW	DESC	Quantity	Rate($/Hr)
MIL	CODEA	Equip. Operator	1.0	21.2
MIL	CODEA	Hyd. Excavator	1.0	35.45
MIL	CODEA	Laborer	1.0	12.35
MIL	CODEA	Small Tools	0.11	1.39

items has been interpreted and expressed in the form of rules. The following rule identifies the MCACES item for excavation with a hydraulic-excavator that has a bucket capacity of 0.5 cu yd* and an excavation rate of 75 cu yd per hour in medium soil.

```
(define-rule mcaces-hydraulic-excavator-rule-1
    (:direction :forward)
    (instance ?query is IDENTIFY-MCACES-KEYS-QUERY)
    (bind ?act-inst (send-msg ?query :activity))
    (instance ?act-inst is EXCAVATION
        with soil-type medium)
    (bind ?cm-inst (send-msg ?act-inst :construction-method))
    (instance ?cm-inst is HYDRAULIC-EXCAVATOR-CM
        with bucket-capacity 0.5 :unit cy
        with rate 75 :unit cy-per-hr)
    THEN
    (instance ?query is IDENTIFY-MCACES-KEYS-QUERY
        with mcaces-keys (('volume ("MIL" "02221" "1202"))))
    (print-out "mcaces-hydraulic-excavator-rule-1 .. fired")
```

Sequencing Construction Activities

Activity sequences are determined on the basis of predefined activity sequences (e.g., for the same group of design components, formwork always precedes reinforcing), relationships between groups of design components, and crew interactions. Considerable work has been conducted in this area. Both OARPLAN (Darwiche 1990) and CASCH (Echeverry 1991) have identified many of these relationships and the associated sequencing rules. This approach distinguishes between two categories of rules: (1) sequencing activities within a component group and (2) sequencing activities belonging to different component groups. The knowledge for sequencing activities within a component group may be organized by type of material used to construct building component, or by general component categories. For example, the following rule expresses the knowledge that for all cast-in-place concrete components, the formwork must be erected before reinforcement is placed.

```
(define-rule cip-concrete-place-reinf-sequence-rule
    (:direction :forward)
    (instance ?query is IDENTIFY-ACTIVITY-SEQUENCE-BY-COMPTYPE-QUERY)
    (bind ?cmpgrp (send-msg ?query :component-group))
    (instance ?cmpgrp is COMPONENT-GROUP)
```

*1 cu yd = 0.7646 m^3.

```
(bind ?compinst (send-msg ?cmpgrp :standard-component-instance))
(instance ?compinst is CIP-CONCRETE)
(instance ?act1 is ERECT-CONCRETE-FORMWORK)
(component-group-for-activity ?act1 ?cmpgrp)
(instance ?act2 is PLACE-CONCRETE-REINFORCEMENT)
(component-group-for-activity ?act2 ?cmpgrp)
(unknown (precedes ?act1 ?act2))
THEN
(precedes ?act1 ?act2)
(print-out "cip-concrete-place-reinf-sequence-rule .. fired")
```

Sequencing of activities in different component groups is based on the "sequencing relationships" between the components identified in earlier work. For example, the following rule expresses the knowledge that if Component Group 1 is supported by Component Group 2, then all activities associated with Component Group 2 must precede all activities associated with Component Group 1.

```
(define-rule supported-by-rule
    (:direction :forward)
    (instance ?query is IDENTIFY-COMPONENT-GROUP-SEQUENCE-QUERY)
    (instance ?cmpgrp1 is COMPONENT-GROUP)
    (instance ?cmpgrp2 is COMPONENT-GROUP)
    (supported-by ?cmpgrp1 ?cmpgrp2)
    (bind ?n1 (send-msg ?cmpgrp1 :get-name))
    (bind ?n2 (send-msg ?cmpgrp2 :get-name))
    THEN
    (precedes ?cmpgrp2 ?cmpgrp1)
    (print-out ?n1 " supported-by" ?n2)
```

At times it may be necessary to establish relationships between specific activities in component groups. For example, when a cast-in-place concrete foundation wall is supported by a footing, backfill activity for the footings is done after the foundation wall formwork is removed. If the foundation wall is a basement wall, then backfill activity for footings is done after the floor-slab it supports is cured. At the present time, spatial and other interactions between crews involved in various activities are not considered. However, this type of knowledge is readily accommodated in the rule-based framework.

Leads or lags also can be specified in the sequencing rules (e.g., between curing concrete and removing formwork). In the absence of such rules, an approximate rule is used to assign leads and lags to precedence relationships. According to this rule, if

activity A1 (with duration D1) is the immediate predecessor of A2 (with duration D2), then lead/lag is assigned subject to following constraints: (1) A2 can start only after 25 percent of A1 has been completed, and (2) rate of completion of A2 must not overtake the rate of completion of A1.[*]

Construction Cost Estimation

As explained in ***Resource Assignment and Activity Duration Computation*** (p 37), the MCACES unit price databases are used to obtain crew definitions, unit material, and labor and equipment costs for activities involved in the installation of common building components. The actual costs may then be computed based on the quantities and unit costs.

Dealing With Incomplete Design Information

An incomplete design may be elaborated by identifying feasible system alternatives for the current design context from the database of prior designs. Some of the advantages of using alternatives from prior designs include: (1) they conform to guidelines established by the U.S. Army Corps of Engineers (Architecture and Engineering Instructions, Design Criteria, 1994) because they are from previous Corps projects, and (2) they will incorporate location-specific factors such as weather conditions and material availability (assuming that in the prior designs the selection of systems would have taken into account such factors). The existence of a database of previously completed as-built facility designs is assumed (also assuming that the database contains "successful" designs [i.e., ones that did not result in problems during construction or operation of the facility]). USACERL researchers have developed a simple object-based language to express queries for retrieving assemblies from the previous projects. For example, the user can retrieve exterior-wall assemblies with a certain minimum r-value. The Backus-Naur Format (BNF) specification for such queries is shown below:

1. <query-exp> := '(' <query-result> <query-variables> <query-predicates> ')' | '(' all-instances <type> ')'
   where <type> is the name of an object class (e.g., exterior-wall)

2. <query-variables> := '(' <query-var-list> ')' | '(' ')'
   <query-var-list> := '(' <var-name> <query-exp> ')' | '(' <var-name> <query-exp> ')' <query-var-list>

*Personal Communication, Stephen McKuzes, Gilbane Construction Co., Chicago, IL.

3. <query-result> := <var-name>

The target variable name must be from one of the <var-name>'s in the <query-variables>. Instances in the scope of this variable name are returned by the query.

4. <query-predicates> := <predicate-exp> | '(' <logical-and-or> <predicate-exp> <query-predicates> ')'

where, <logical-and-or> := and | or

5. <predicate-exp> := <var-exp> | '(' <op> <var-exp> <value> ')' | '(' <op> <var-exp> <var-exp> ')'

where, <op> := = | >= | <= | > | < | eq | string-equal

Operators can be extended to handle comparisons between other value types such as date. In the current implementation they are limited to above types.

<value> := number | string | symbol

The <value>'s are atomic in that they are simple types and cannot be objects or instances. In the current implementation, they are restricted to the three types indicated above.

6. <var-exp> := '(' <var-name> <path-exp> ')'

The <var-name> is one of the variables specified in <query-variables>. The <path-exp> is describes the traversal path from the variable name ending in a slot-value or call to a method.

7. <path-exp> := '(' [<intermediate-path-item-list>] <end-path-item> ')'

where,

<end-path-item> := :slot <slot-name> | :method <method-name> ['(' <input-args-list> ')']

<input-args-list> := <var-exp> | <input-args-list> <var-exp>

<intermediate-path-item-list> := <intermediate-path-item>
| <intermediate-path-item-list> <intermediate-path-item>

<intermediate-path-item> := :slink <link-type> [<link-data>]

An example of a query using this syntax is:

```
(w ((w (all-instances exterior-wall)))
    (and (eq (w :slink has-part exterior-skin :method :type) brick)
        (eq (w :slink has-part exterior-skin :method :color) red)))
```

This query will retrieve all instances of exterior-wall with an exterior-skin of red brick. The user is responsible for selecting and adapting the selected alternative to the current design context.

5 Current Implementation

Implementation Environment

The Goldworks (GoldWorks User's Manual 1989) expert system development environment was used to develop the prototype application because this was the environment used to build ACE. This prototype functions as an agent within ACE. The DDE (Dynamic Data Exchange) interface ACE has with AutoCAD™ is used to display the building components in AutoCAD™.* An important step in creating a schedule is to use CPM to compute early and late start times to determine the total duration of the construction project. Because many commercial scheduling programs exist that perform CPM and much more, it was decided to integrate one such program (Microsoft® Project) with this environment to provide CPM capability. To present the cost information generated by the system in a convenient spreadsheet format for subsequent manipulation and printing, it was decided to integrate a commercial spreadsheet program (Microsoft® Excel). The Goldworks Open Data Base Connectivity (ODBC) interface was used to access historical project information on crews, productivity, and unit costs from MCACES dBase files (MCACES GOLD User's Manual 1992). Figure 12 illustrates the integration with ACE and other existing commercial software achieved in the prototype.

The Schedule Generation Process

The schedule generation process currently used has the following steps: (1) defining component groups where the "work breakdown structure" is defined by grouping individual building components (see Figure 13) based on material and size attributes of the component; (2) generating activities where activities required to install various components are identified, and parameters and quantities are also computed; (3) identifying construction methods; (4) looking up crew, productivity, and cost information from MCACES historical databases; (5) sequencing activities; and (6) sending the generated schedule to Microsoft® Project (see Figure 14).

---

*AutoDesk, Inc., 111 McInnis Parkway, San Rafael, CA 94903.

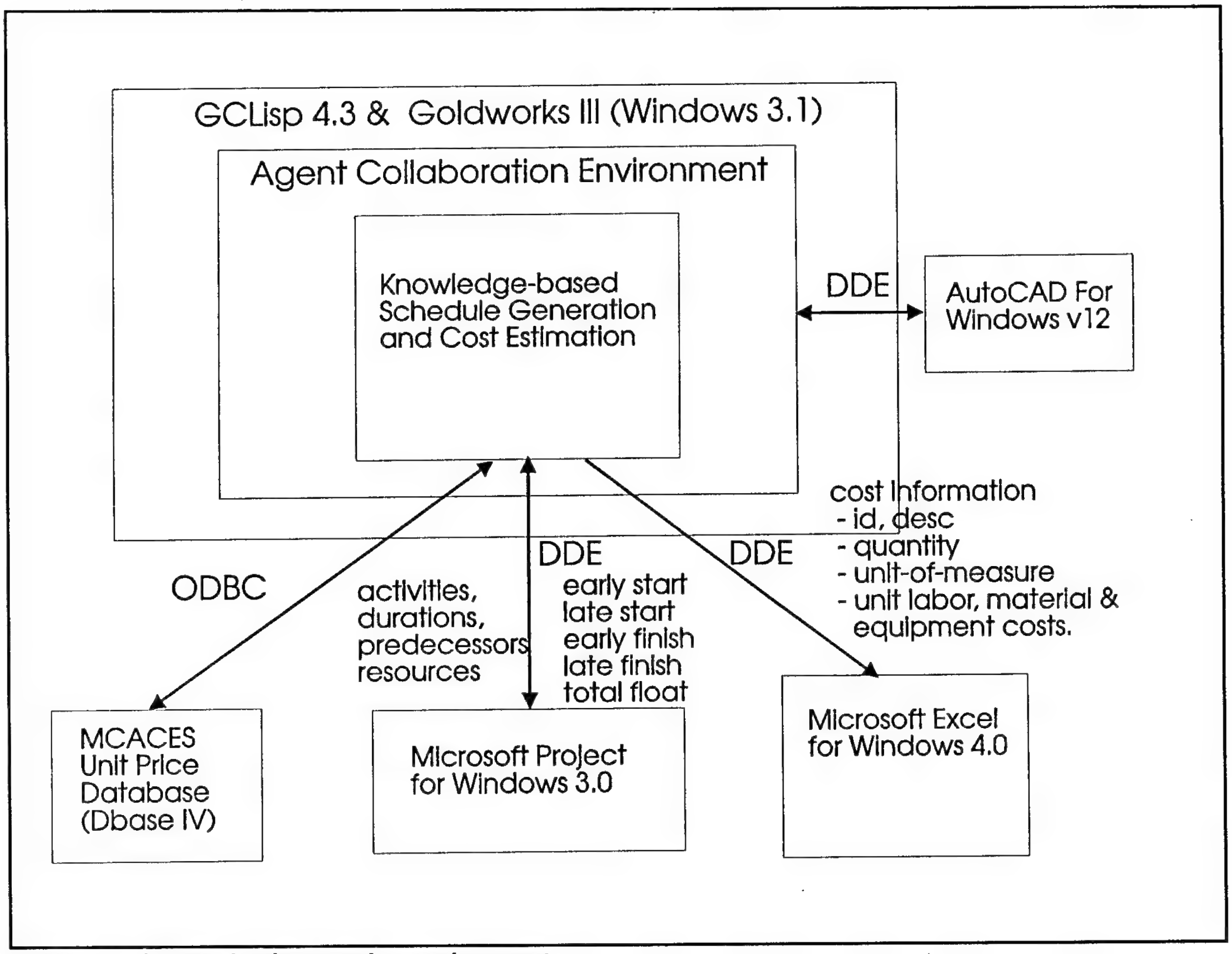


Figure 12. Current implementation environment.

Preliminary schedules and estimates can be generated for comparing alternative design solutions (for example, to consider the effect on the schedule and estimate of using concrete masonry unit interior partitions vs. using metal studs and gypsum wallboard as in Figure 15). Such comparisons do not require regeneration of the entire schedule every time. Based on the changes made to the design, the affected portion of the schedule is identified, and only that portion is regenerated.

Visualizing the Construction Schedule

Commercially available tools such as AutoCAD™ and Walkthru™ provide the graphical interfaces necessary to visualize the construction process. The critical input for these tools is the link between the design components and construction schedule activities. This information is provided by the schedule generation process. A schedule visualization algorithm has been developed to visualize the construction schedule in AutoCAD™. The animation is based on percent complete of building components. The percent complete of components is represented by using different colors (currently

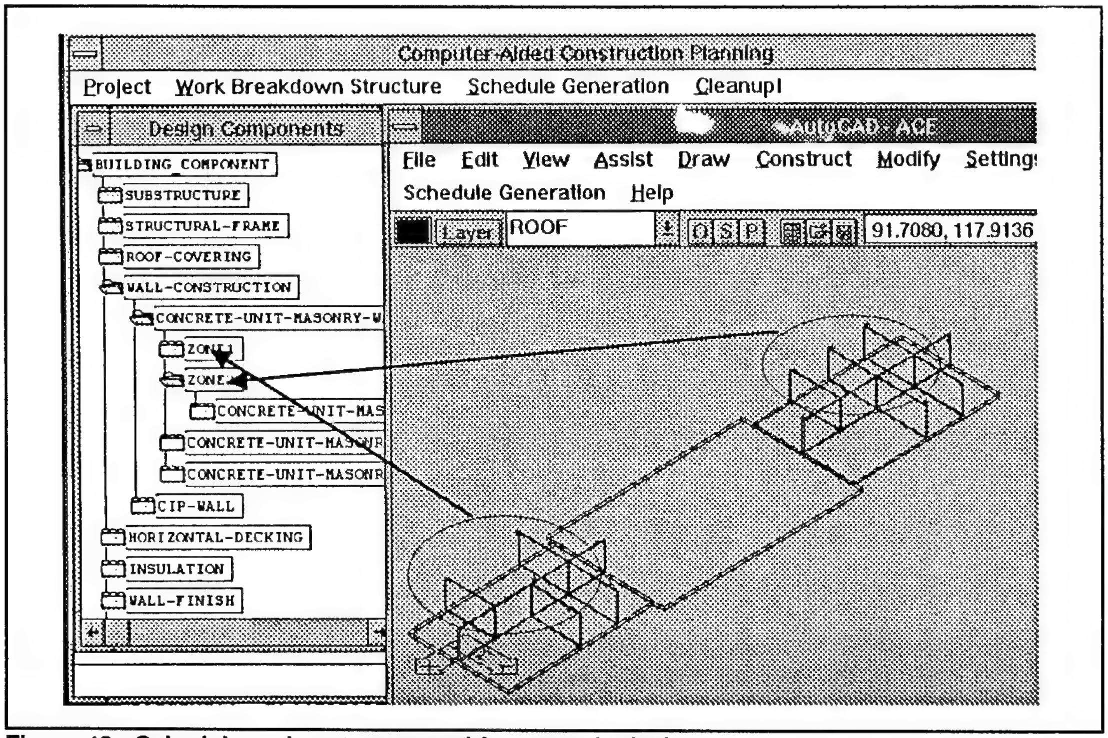


Figure 13. Schedule and cost generated for example design.

seven colors) in AutoCAD™ for Windows™, while the activities associated with the components are highlighted in Microsoft® Project simultaneously (see Figure 16). The visualization process helps verify the correctness of the schedule and also makes identification of any sequencing problems easy so they can be corrected before construction begins.

Scope of Knowledge in the System

The prototype system can manipulate the following kinds of components and assemblies: (1) Foundation: Cast-in-place continuous footings; (2) Foundation Wall: Concrete-masonry-unit or cast-in-place concrete with polyethylene vapor retarder; (3) Floor construction: Standard slab-on-grade; (4) Exterior Wall—Construction: Concrete-masonry-unit or metal-stud; Exterior-skin; Insulation: urethane; (5) Interior Wall—Construction: Concrete-masonry-unit partitions, and (6) Roof—Construction: prefabricated wood trusses or open web joists; Covering: membrane or shingles, and urethane or fiberglass insulation. The intercomponent relationships used for sequencing include: supported-by, weather-protected-by, covered-by, and enclosed-by.

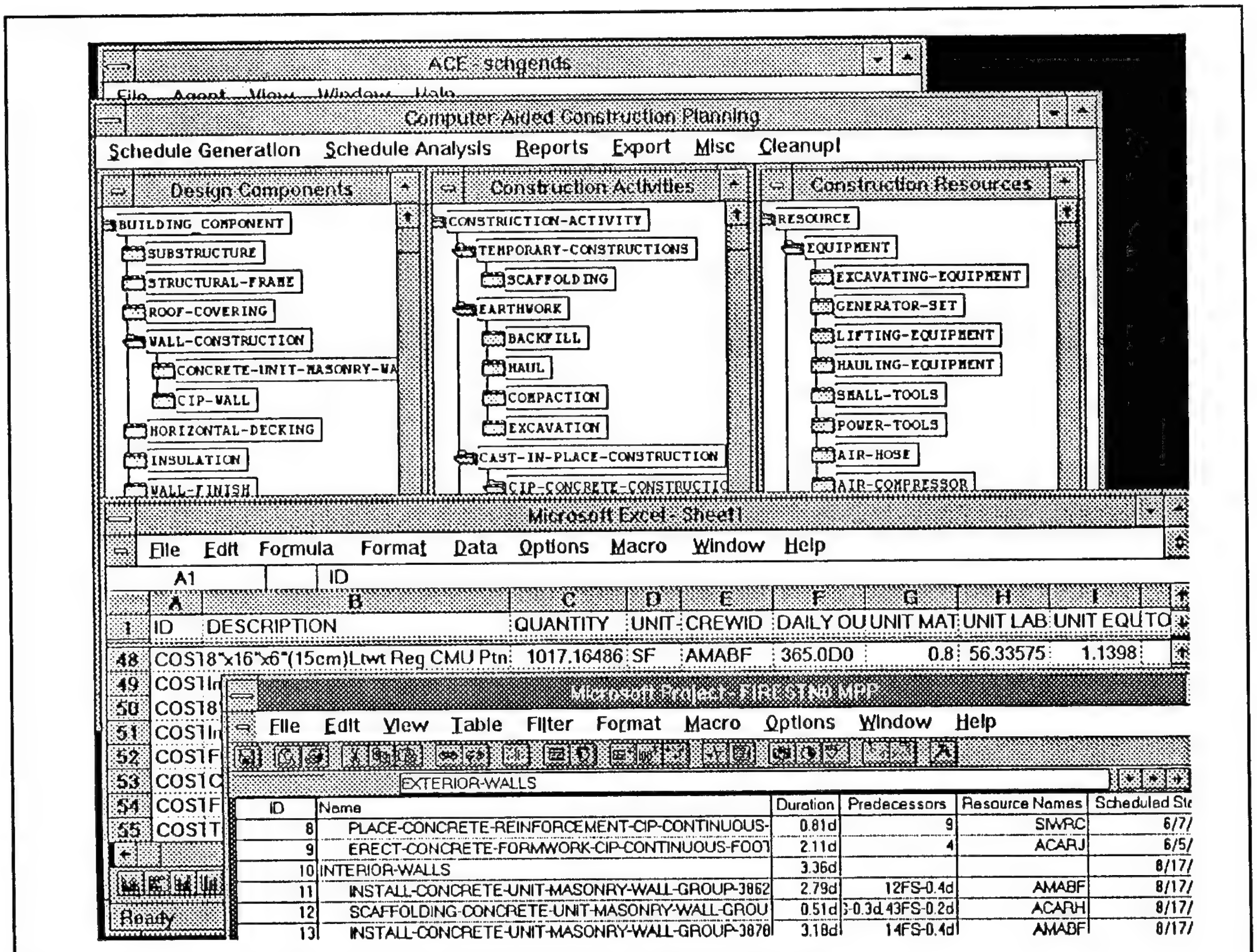


Figure 14. Aggregating components by zone in the prototype implementation.

Extending the Scope: Adding New Knowledge

The system can be extended to accept new kinds of building components or assemblies. To extend the scope, follow these steps:

1. Define attributes and operations and add the new component to the existing building components hierarchy
2. Define new activity classes and add rules to identify activities (including supporting activities) for installation of the new component class (if necessary)
3. Define rules to compute quantities of work for the installation of the new component class (if necessary).
4. Define rules to identify items in MCACES historical cost database (or some other historical cost/productivity database that is being used).
5. Identify any new sequencing relationships and define rules to sequence those relationships (if necessary).

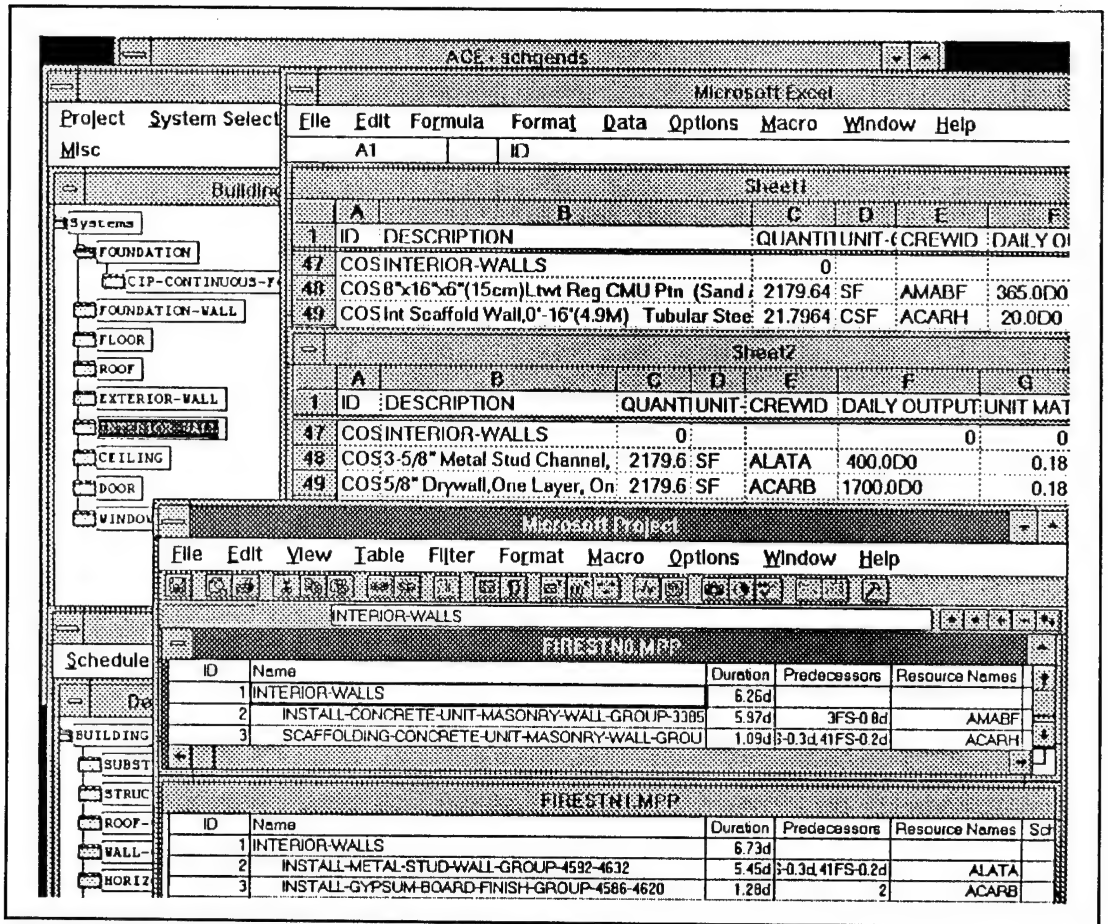


Figure 15. Comparing design alternatives with regard to time and cost.

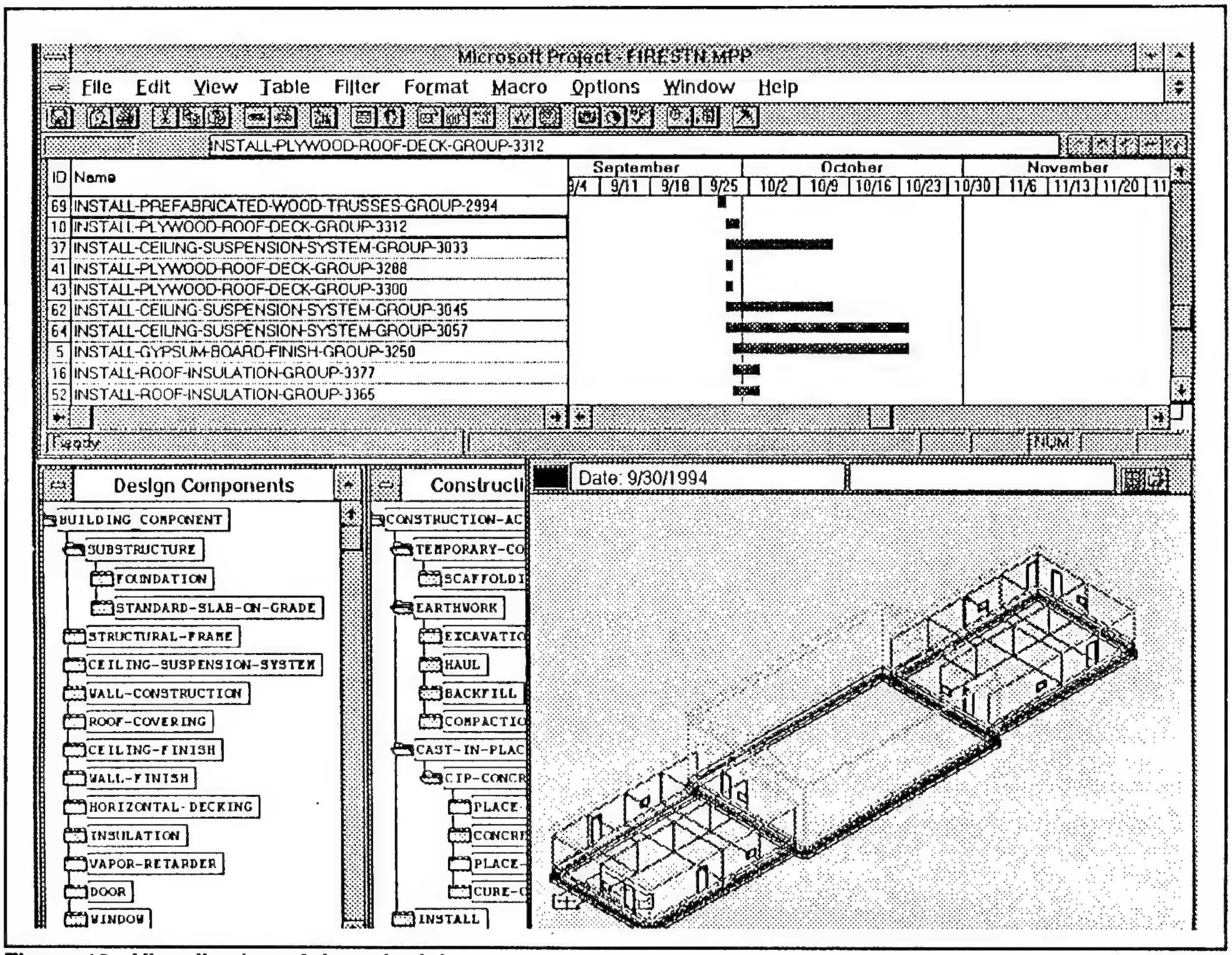


Figure 16. Visualization of the schedule.

7 Conclusions and Future Enhancements

Conclusions

USACERL researchers developed a methodology using object-oriented and rule-based concepts to generate a preliminary construction plan for facility designs that will:

- compare alternative designs from a construction time and cost perspective
- animate the schedule
- provide a good baseline schedule and cost estimate for the evaluation of contractor bids
- determine the impact on schedule and costs due to change orders and modifications.

This methodology enables iterative processing of construction plans that may be generated and evaluated several times during the course of planning, designing, and constructing a facility. This research goes further than previous approaches in that it considers all aspects of the planning process based on intercomponent relationships. The concept of a "component-group" was introduced as an intermediate abstraction to relate activities to design components, thus integrating the product model (design) with the process (schedule). Demonstrations of the prototype have indicated that this corresponds to actual practice employed by construction planners.

Since the generation of a preliminary schedule is automated, this approach allows user interaction for analysis and modification of the schedule and design. A construction planner using this tool during preliminary design would be able to provide feedback to designers and others earlier in the design process than is typical for most projects. Also, animation of a baseline schedule generated using the Construction Planning Agent would be very useful during discussions with the contractor to evaluate teade coordination, constructibility, and feasibility of contractor's schedule submittals, schedule impact due to change orders, etc.

The prototype system was demonstrated to construction schedulers and estimators both within DOD (USACERL and USACE) and in the private sector (W.E. O'Neil Construction, Stone & Webster Engineering Corporation, Bechtel Corporation, and Gilbane Building Company) and was well received.

Future Enhancements

Based on feedback from these demonstrations, the following additions and improvements are being considered:

1. Incorporate capability to consider weather impacts on construction plans (Steen 1991)
2. Develop capability to generate and modify crew sizes to evaluate cost and time tradeoffs
3. Facilitate greater flexibility in modeling construction methods by incorporating a multilevel activity generation process (i.e., after a first level of activities are generated, construction methods are selected, and, based on the choice of the method, a second level of activities are generated)
4. Investigate ways of automating the generation of intercomponent relationships (e.g., supported-by)
5. Develop knowledge-acquisition capabilities so users can add or modify the knowledge base and extend the system
6. Enhance the capability to deal with incomplete design information by automating the selection of system alternatives and their adaption to the current design context.

This effort is part of the USACERL "Construction CADD" research project, which has the long-term goals to:

1. Use intelligent design information as much as possible for BCO (biddability, constructability, and operability) reviews, cost estimating, scheduling, project control, quality assurance, and capturing as-built information.
2. Minimize redundant data input by starting with electronic design information, tracking changes throughout construction, and adding actual component information during construction to create more accurate as-builts (intelligent CADD drawings and associated files).
3. Create a detailed object-oriented CADD model to support construction progress, monitoring, and control. This graphical representation will be linked to relevant objects developed during research being done as part of the Collaborative Engineering project. The 3-D representation of construction progress will provide the capability to model alternative construction procedures and methods; track actual construction progress in real time; and support project management decisionmaking.
4. Build an integrated information framework to allow use of intelligent design information, addition of detailed building component data, project control information, and multimedia as-built information during construction and delivery

of a CD-ROM to the owner of the constructed facility. The resulting framework will support the delivery of a complete "audit" train of all pertinent building component information, including parts, vendors, maintenance and repair inspection schedules, and as-built and as-installed drawings.

References

ACE 1.1. Developer's Manual (U.S. Army Construction Engineering Research Laboratories [USACERL], 1995).

Ayman, M.A., *Geometric-Based Reasoning System for Project Planning Utilizing AI and CAD Technologies*, Ph.D. thesis (Virginia Polytechnic Institute, 1992).

Beach, James, *Preparing, Using and Analyzing the Project Schedule*, (American Society of Civil Engineers [ASCE] Seminar Videotape Program, 1993).

Darwiche, A., R.E. Levitt, and B. Hayes-Roth, *OARPLAN: Generating Project Plans in a Blackboard System by Reasoning about Objects, Actions and Resources*, Technical Report No. 2 (Center for Integrated Facility Engineering, Stanford University, 14 February 1989).

Echeverry, D., *Factors for Generating Initial Construction Schedules*, Ph.D. dissertation (Department of Civil Engineering, University of Illinois at Urbana-Champaign, 1990).

Fikes, R.E., "STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving," *Artificial Intelligence*, Vol. 2, No. 4 (1971).

Golish, M. "Architect's Associate: Applying Concurrent Engineering to Facility Design," *Research, Development, and Acquisition* (25 October 1993).

Halpin, D.W., and R.W. Woodhead, *Design of Construction and Process Operations* (John Wiley & Sons, 1976).

Heckel, J., K. McGraw, and M. Case, "A Methodology for Linking Symbolic and Graphical Models for Collaborative Engineering" DRAFT Technical Report (USACERL, 1995).

Jones, A., and D. Clegg, *CI / SfB Construction Indexing Manual* (RIBA Publications, 1976).

Kim, Jae-Jun, *A Work-Package Based Process Model for Petrochemical Construction Planning*, Ph.D. thesis (Department of Civil Engineering, University of Illinois at Urbana-Champaign, 1993).

MCACES GOLD User's Manual, Version 5.2 (Building Systems Design, Inc., 1992).

MEANS Building Construction Cost Data, R.S. Means Company, Kingston, MA, 1992.

Nomani, Z., Nie-Jia Yau, D. Echeverry, A. Hassanein, and J.W. Melin, *Activity Definition Index*. Working Document (Construction Engineering and Management, University of Illinois at Urbana-Champaign, 1990).

Rumbaugh, J., M. Blaha, W. Premerlani, F. Eddy, and W. Lorensen, *Object-Oriented Modeling and Design* (Prentice Hall, 1991).

Sacerdoti, E.D., *A Structure for Plans and Behavior* (Elsevier-Holland, 1977).

Steen, S.L., D. Echeverry, M. Aboushousha, and S. Kim, *Severe Weather Impact Analysis for Military Construction Projects*, Interim Report P-91/50/ADA242394 (USACERL, September 1991).

Tatum, C.B., J.A. Vanegas, and J.M. Williams, *Constructability Improvement Using Prefabrication, Preassembly, and Modularization*, Technical Report No. 297 (Department of Civil Engineering, Stanford University, November 1986).

Thomas, H.R., and G.R. Smith, *Loss of Construction Labor Productivity Due to Inefficiencies and Disruptions*, Technical Report (The Pennsylvania Transportation Institute, 1990).

Uniformat, "*A Uniform Classification of Construction Systems and Assemblies*," Interim edition (Construction Specifications Institute, 1992).

Winstanley, G., and K. Hoshi, *Model-Based Planning Utilizing Activity Aggregation Based on Zones*, Technical Report #69 (Center for Integrated Facility Engineering, Stanford University, 1992).

Winstanley, G., M.A. Chacon, and R.E. Levitt, "Model-Based Planning: Scaled-Up Construction Application," *Journal of Computing in Civil Engineering*, Vol 7, No. 2 (April 1993).

Yau, Nie-Jia, *An Object-Oriented Project Model for Integrating Building Design, Construction Scheduling, and Cost Estimating for Mid-Rise Construction*, Ph.D. dissertation (Department of Civil Engineering, University of Illinois, Urbana-Champaign, 1992).

Zozoya-Gorostiza, C., C. Hendrickson, and D. Rehak, *Knowledge-Based Process Planning for Construction and Manufacturing* (Academic Press, London, 1989).

DISTRIBUTION

Chief of Engineers
ATTN: CEHEC-IM-LH (2)
ATTN: CEHEC-IM-LP (2)
ATTN: CECC-R
ATTN: CERD-L
ATTN: CEMP-C
ATTN: CEMP-CE (2)
ATTN: CEMP-CP (3)
ATTN: CEMP-E
ATTN: CEMP-EA (3)
ATTN: CEMP-ES (2)

US Army Engr District
ATTN: Library (40)
ATTN: Chief/Project Management (40)
ATTN: Civil Construction/Civil Con-Ops (40)
ATTN: Design Branch (40)
Sacramento District
ATTN: ARMS TCX (2)

US Army Engr Division
ATTN: Library (11)
ATTN: Chief/Project Management (11)
ATTN: Civil Construction/Civil Con-Ops (11)
ATTN: Design Division (11)

Defense Tech Info Center 22304
ATTN: DTIC-FAB (2)

226
9/95